UNIVERSIDAD COMPLUTENSE DE MADRID
FACULTAD DE INFORMÁTICA
DEPARTAMENTO DE INGENIERÍA DEL SOFTWARE E INTELIGENCIA ARTIFICIAL

TESIS DOCTORAL

Extraction and Analysis of Features for Identifying, Clustering and Modifying the Source of Images Generated by Mobile Devices

Extracción y Análisis de Características para Identificación, Agrupamiento y Modificación de la Fuente de Imágenes Generadas por Dispositivos Móviles

MEMORIA PARA OPTAR AL GRADO DE DOCTOR

PRESENTADA POR

Jocelin Rosales Corripio

DIRECTORES

Luis Javier García Villalba
Ana Lucila Sandoval Orozco

Madrid, 2018

Extracción y Análisis de Características para Identificación, Agrupamiento y Modificación de la Fuente de Imágenes Generadas por Dispositivos Móviles

Extraction and Analysis of Features for Identifying, Clustering and Modifying the Source of Images Generated by Mobile Devices

Thesis by

Jocelin Rosales Corripio

In Partial Fulfillment of the Requirements for the Degree of
Doctor por la Universidad Complutense de Madrid en el
Programa de Doctorado en Ingeniería Informática

Advisors

Luis Javier García Villalba

Ana Lucila Sandoval Orozco

Facultad de Informática
Universidad Complutense de Madrid

Madrid, April 2017

Extraction and Analysis of Features for Identifying, Clustering and Modifying the Source of Images Generated by Mobile Devices

Thesis by

Jocelin Rosales Corripio

In Partial Fulfillment of the Requirements for the Degree of
Doctor por la Universidad Complutense de Madrid en el
Programa de Doctorado en Ingeniería Informática

Advisors

Luis Javier García Villalba

Ana Lucila Sandoval Orozco

Facultad de Informática
Universidad Complutense de Madrid

Madrid, April 2017

Extracción y Análisis de Características para Identificación, Agrupamiento y Modificación de la Fuente de Imágenes Generadas por Dispositivos Móviles

TESIS DOCTORAL

Memoria presentada para obtener el título de
Doctor por la Universidad Complutense de Madrid
en el Programa de Doctorado en Ingeniería Informática

Jocelin Rosales Corripio

Directores

Luis Javier García Villalba
Ana Lucila Sandoval Orozco

Facultad de Informática
Universidad Complutense de Madrid

Madrid, Abril de 2017

Dissertation submitted by Jocelin Rosales Corripio to the *Faculty of Computer Science and Engineering* of the *Universidad Complutense de Madrid* in Partial Fulfillment of the Requirements for the Degree of *Doctor por la Universidad Complutense de Madrid en el Programa de Doctorado en Ingeniería Informática*.

Madrid, 2017.

(Submitted April 20, 2017)

Title:

Extraction and Analysis of Features for Identifying, Clustering and Modifying the Source of Images Generated by Mobile Devices

PhD Student:

Jocelin Rosales Corripio (jocelinr@ucm.es)
Departamento de Ingeniería del Software e Inteligencia Artificial
Facultad de Informática
Universidad Complutense de Madrid
28040 Madrid, Spain

Advisors:

Luis Javier García Villalba (javiergv@fdi.ucm.es)
Ana Lucila Sandoval Orozco (asandoval@fdi.ucm.es)

This work has been done within the Group of Analysis, Security and Systems (GASS, `http://gass.ucm.es/`), Research Group 910623 from the Universidad Complutense de Madrid (UCM) as part of the activities of the research project funded by the European Commission Horizon 2020 Programme under Grant Agreement number H2020-FCT-2015/700326-RAMSES (Internet Forensic Platform for Tracking the Money Flow of Financially Motivated Malware). This research has also been supported by Ayudas de Postgrado Santander - Convocatoria ECL 2012. Part of this work was done during my stay in Mexico at Benemérita Universidad de Autónoma de Puebla (Faculty of Computer Science). Part of the computations of this work were performed in Eolo, the HPC of Climate Change of the International Campus of Excellence of Moncloa, funded by the Ministerio de Educación, Cultura y Deporte (MECD, Spain) and the Ministerio de Ciencia e Innovación (MICINN, Spain). This is a contribution to CEI Moncloa.

Tesis Doctoral presentada por la doctoranda Jocelin Rosales Corripio en la Facultad de Informática de la Universidad Complutense de Madrid para la obtención del título de Doctor por la Universidad Complutense de Madrid en el Programa de Doctorado en Ingeniería Informática.

Terminada en Madrid el 20 de Abril de 2017.

Título:

Extracción y Análisis de Características para Identificación, Agrupamiento y Modificación de la Fuente de Imágenes Generadas por Dispositivos Móviles

Doctoranda:

Jocelin Rosales Corripio (jocelinr@ucm.es)
Departamento de Ingeniería del Software e Inteligencia Artificial
Facultad de Informática
Universidad Complutense de Madrid
28040 Madrid, España

Directores:

Luis Javier García Villalba (javiergv@fdi.ucm.es)
Ana Lucila Sandoval Orozco (asandoval@fdi.ucm.es)

Esta tesis doctoral ha sido realizada dentro del grupo de investigación GASS (Grupo de Análisis, Seguridad y Sistemas, grupo 910623 del catálogo de grupos reconocidos por la UCM) como parte de las actividades del proyecto de investigación RAMSES (Internet Forensic Platform for Tracking the Money Flow of Financially-Motivated Malware) financiado por la Comisión Europea dentro del Programa Marco de Investigación e Innovación Horizonte 2020 (H2020-FCT-2015/700326-RAMSES). Asimismo, el presente trabajo ha sido financiado por el Banco Santander a través de las Ayudas de Postgrado Santander Convocatoria ECL 2012. Parte de esta investigación ha sido realizada en la Facultad de Ciencias de la Computación de la Benemérita Universidad Autónoma de Puebla (BUAP). Parte de los cálculos de este trabajo fueron realizados en EOLO, el sistema de computación de alto rendimiento del Clúster de Cambio Global y Nuevas Energías del Campus de Excelencia Internacional (CEI) Campus Moncloa, financiado por el Ministerio de Educación, Cultura y Deporte y por el Ministerio de Ciencia e Innovación. Esto es una contribución al CEI Moncloa.

This thesis is dedicated to my family, especially to my son.

Esta tesis está dedicada a mi familia, en especial a mi hijo.

Acknowledgments

Every day of my life I will be grateful to have Javier the best life partner. Thanks for always being by my side helping me to achieve any dream I have no matter how small.

To my parents (including Judith) and brothers, for all the unconditional support they give me every day in all aspects to help me be happy.

To Javier García and Ana Lucila for the impulse, the guidance, the support, the trust, and the friendship they have given me all these years. Thank you both for allowing me to achieve this goal without losing others that are also important. Especially to Ana for her invaluable help to achieve this work.

This work has been done within the Group of Analysis, Security and Systems (GASS, `http://gass.ucm.es/`), Research Group 910623 from the Universidad Complutense de Madrid (UCM) as part of the activities of the research project funded by the European Commission Horizon 2020 Programme under Grant Agreement number H2020-FCT-2015/700326-RAMSES (Internet Forensic Platform for Tracking the Money Flow of Financially Motivated Malware). This research has also been supported by Ayudas de Postgrado Santander - Convocatoria ECL 2012. Part of this work was done during my stay in Mexico at Benemérita Universidad de Autónoma de Puebla (Faculty of Computer Science). Part of the computations of this work were performed in Eolo, the HPC of Climate Change of the International Campus of Excellence of Moncloa, funded by the Ministerio de Educación, Cultura y Deporte (MECD, Spain) and the Ministerio de Ciencia e Innovación (MICINN, Spain). This is a contribution to CEI Moncloa.

Agradecimientos

Cada día de mi vida estaré agradecida por tener a Javier, el mejor compañero de vida. Gracias por estar siempre a mi lado y ayudarme a lograr cualquier sueño que tenga por pequeño que sea.

A mis padres (incluyendo a Judith) y hermanos por todo el apoyo incondicional que me dan día a día en todos los aspectos para ayudarme a ser feliz.

A Javier García y a Ana Lucila por el impulso, la guía, el apoyo, la confianza, y la amistad que me han brindado todos estos años. Gracias a los dos por permitirme lograr esta meta sin perder otras que también son importantes. En especial a Ana por su inestimable ayuda para lograr este trabajo.

Esta tesis doctoral ha sido realizada dentro del grupo de investigación GASS (Grupo de Análisis, Seguridad y Sistemas, grupo 910623 del catálogo de grupos reconocidos por la UCM) como parte de las actividades del proyecto de investigación RAMSES (Internet Forensic Platform for Tracking the Money Flow of Financially-Motivated Malware) financiado por la Comisión Europea dentro del Programa Marco de Investigación e Innovación Horizonte 2020 (H2020-FCT-2015/700326-RAMSES). Asimismo, el presente trabajo ha sido financiado por el Banco Santander a través de las Ayudas de Postgrado Santander Convocatoria ECL 2012. Parte de esta investigación ha sido realizada en la Facultad de Ciencias de la Computación de la Benemérita Universidad Autónoma de Puebla (BUAP). Parte de los cálculos de este trabajo fueron realizados en EOLO, el sistema de computación de alto rendimiento del Clúster de Cambio Global y Nuevas Energías del Campus de Excelencia Internacional (CEI) Campus Moncloa, financiado por el Ministerio de Educación, Cultura y Deporte y por el Ministerio de Ciencia e Innovación. Esto es una contribución al CEI Moncloa.

Contents

List of Figures

List of Tables

List of Algorithms

List of Acronyms

1NN	*First Nearest Neighbor*
ADC	*Analog Digital Converter*
CCD	*Charge Coupled Device*
CCFL	*Cold Cathode Fluorescent Lamp*
CFA	*Color Filter Array*
CIS	*Contact Image Sensors*
CMOS	*Complementary Metal Oxide Semiconductor*
CYGM	*Cyan-Yellow-Green-Magenta*
CYYM	*Cyan-Yellow-Yellow-Magenta*
DCT	*Discrete Cosine Transform*
DFT	*Discrete Fourier Transform*
DIP	*Digital Image Processor*
DSC	*Digital Still Camera*
EAM	*Error Absoluto Medio*
EC	*Error Cumulants*
ECM	*Error Cuadrático Medio*
ERE	*Eigenfeature Regularization*
Exif	*Exchangeable Image File Format*

FPN *Fixed Pattern Noise*

GPS *Global Positioning System*

GRGB *Green-Red-Green-Blue*

HVS *Human Visual System*

IQM *Image Quality Metrics*

JPEG *Joint Photographic Experts Group*

LMSE *Laplacian Mean Square Error*

MAE *Mean Absolute Error*

MOS *Metal Oxide Semiconductor*

MSE *Mean Square Error*

ND *Non Detailed*

NGS *Normalized Group Sizes*

PMT *Photomultiplier Tubes*

PNU *Pixel Non-Uniformity*

PRNU *Photo Response Non-Uniformity*

PSVM *Probabilistic Support Vector Machine*

QMF *Quadrature Mirror Filter*

RGB *Red-Green-Blue*

RGBE *Red-Green-Blue-Emerald*

RGBW *Red-Green-Blue-White*

ROI *Region of Interest*

SFFS *Sequential Floating Forward Selection*

SPN *Sensor Pattern Noise*

SVM *Support Vector Machine*

TPR *True Positive Rate*

Abstract

Nowadays, digital images play an important role in our society. The presence of mobile devices with integrated cameras is growing at an unrelenting pace, resulting in the majority of digital images coming from this kind of device. Technological development not only facilitates the generation of these images, but also the malicious manipulation of them. Therefore, it is of interest to have tools that allow the device that has generated a certain digital image to be identified. The digital image source can be identified through the features that the generating device permeates it with during the creation process. In recent years most research on techniques for identifying the source has focused solely on traditional cameras. The forensic analysis techniques of digital images generated by mobile devices are therefore of particular importance since they have specific characteristics which allow for better results, and forensic techniques for digital images generated by another kind of device are often not valid.

This thesis provides various contributions in two of the main research lines of forensic analysis, the field of identification techniques and the counter-forensics or attacks on these techniques. In the field of digital image source acquisition identification techniques, both closed and open scenarios are addressed.

In closed scenarios, the images whose acquisition source are to be determined belong to a group of devices known a priori. Meanwhile, an open scenario is one in which the images under analysis belong to a set of devices that is not known a priori by the forensic analyst. In this case, the objective is not the concrete image acquisition source identification, but their classification into groups whose images all belong to the same mobile device. The image clustering techniques are of particular interest in real situations since in many cases the forensic analyst does not know a priori which devices have generated certain images.

Firstly, techniques for identifying the device type (computer, scanner or digital camera of the mobile device) or class (make and model) of the image acquisition source in mobile devices are proposed, which are two relevant branches of forensic analysis of mobile device images. An approach based on different types of image features and Support Vector Machine as a classifier is presented.

Secondly, a technique for the identification in open scenarios that consists of grouping digital images of mobile devices according to the acquisition source is developed, that is to say, a class-grouping of all input images is performed. The proposal is based on the combination of hierarchical grouping and flat grouping using the Sensor Pattern Noise.

Lastly, in the area of attacks on forensic techniques, topics related to the robustness of the image source identification forensic techniques are addressed. For this, two new algorithms based on the sensor noise and the wavelet transform are designed, one for the destruction of the image identity and another for its forgery. Results obtained by the two algorithms were compared with other tools designed for the same purpose. It is worth mentioning that the solution presented in this work requires less amount and complexity of input data than the tools to which it was compared.

Finally, these identification techniques have been included in a tool for the forensic analysis of digital images of mobile devices called Theia. Among the different branches

of forensic analysis, Theia focuses mainly on the trustworthy identification of make and model of the mobile camera that generated a given image. All proposed algorithms have been implemented and integrated in Theia thus strengthening its functionality.

Keywords: Acquisition Source Identification, Classification, Digital Image, Forensics Analysis, Image Anonymity, Clustering, Image Features, Image Forgery, Mobile Device, Photo Response Non Uniformity, Source, Sensor Noise, Support Vector Machine, Theia, Wavelet Transform.

Resumen

Actualmente las imágenes digitales desempeñan un papel importante en nuestra sociedad. La presencia de dispositivos móviles con cámaras fotográficas integradas crece a un ritmo imparable, provocando que la mayoría de las imágenes digitales procedan de este tipo de dispositivos. El desarrollo tecnológico no sólo facilita la generación de estas imágenes, sino también la manipulación malintencionada de éstas. Es de interés, por tanto, contar con herramientas que permitan identificar al dispositivo que ha generado una cierta imagen digital. La fuente de una imagen digital se puede identificar a través de los rasgos que el dispositivo que la genera impregna en ella durante su proceso de creación. La mayoría de las investigaciones realizadas en los últimos años sobre técnicas de identificación de la fuente se han enfocado únicamente en las cámaras tradicionales. Las técnicas de análisis forense de imágenes generadas por dispositivos móviles cobran, pues, especial importancia, ya que éstos presentan características específicas que permiten obtener mejores resultados, no siendo válidas muchas veces además las técnicas forenses para imágenes digitales generadas por otros tipos de dispositivos.

La presente Tesis aporta diversas contribuciones en dos de las principales líneas del análisis forense: el campo de las técnicas de identificación de la fuente de adquisición de imágenes digitales y las contramedidas o ataques a estas técnicas. En el primer campo se abordan tanto los escenarios cerrados como los abiertos. En el escenario denominado cerrado las imágenes cuya fuente de adquisición hay que determinar pertenecen a un grupo de dispositivos conocidos a priori. Por su parte, un escenario abierto es aquel en el que las imágenes pertenecen a un conjunto de dispositivos que no es conocido a priori por el analista forense. En este caso el objetivo no es la identificación concreta de la fuente de adquisición de las imágenes, sino su clasificación en grupos cuyas imágenes pertenecen todas al mismo dispositivo móvil. Las técnicas de agrupamiento de imágenes son de gran interés en situaciones reales, ya que en muchos casos el analista forense desconoce a priori cuáles son los dispositivos que generaron las imágenes.

En primer lugar se presenta una técnica para la identificación en escenarios cerrados del tipo de dispositivo (computador, escáner o cámara digital de dispositivo móvil) o la marca y modelo de la fuente en dispositivos móviles, que son dos problemáticas relevantes del análisis forense de imágenes digitales. La propuesta muestra un enfoque basado en distintos tipos de características de la imagen y en una clasificación mediante máquinas de soporte vectorial.

En segundo lugar se diseña una técnica para la identificación en escenarios abiertos que consiste en el agrupamiento de imágenes digitales de dispositivos móviles según la fuente de adquisición, es decir, se realiza un agrupamiento en clases de todas las imágenes de entrada. La propuesta combina agrupamiento jerárquico y agrupamiento plano con el uso del patrón de ruido del sensor.

Por último, en el área de los ataques a las técnicas forenses se tratan temas relacionados con la robustez de las técnicas forenses de identificación de la fuente de adquisición de imágenes. Se especifican dos algoritmos basados en el ruido del sensor y en la transformada wavelet; el primero destruye la identidad de una imagen y el segundo

falsifica la misma. Los resultados obtenidos por estos dos algoritmos se comparan con otras herramientas diseñadas para el mismo fin, observándose que la solución aquí presentada requiere de menor cantidad y complejidad de datos de entrada. Finalmente, estas técnicas de identificación han sido incluidas en una herramienta para el análisis forense de imágenes digitales de dispositivos móviles llamada Theia. Entre las diferentes ramas del análisis forense, Theia se centra principalmente en la identificación confiable de la marca y el modelo de la cámara móvil que generó una imagen dada. Todos los algoritmos desarrollados han sido implementados e integrados en Theia, reforzando así su funcionalidad.

Palabras clave: Agrupamiento, Análisis Forense, Anonimización de Imágenes, Características de la Imagen, Clasificación, Dispositivo Móvil, Falsificación, Fuente, Identificación de la Fuente de Adquisición, Imagen Digital, Máquinas de Soporte Vectorial, Respuesta Fotónica No Uniforme, Ruido del Sensor, Theia, Transformada Wavelet.

Part I

Description of the Research

Chapter 1

Introduction

Frequently photographs are considered valuable evidence of the truth as they are real facts captured by electronic devices (cameras). However, with the development of powerful technology and sophisticated tools, it is easier to modify digital images, even for those who do not have technical or specialized knowledge in the area [GKWB07]. The development of digital technologies continues progressing at an unstoppable pace. Every day the number of digital cameras is increasing, as well as the ease of access to them. Since the year 2000 when the first camera phone was introduced in the market, the number of mobile users has increased fivefold. For the year 2020, there will be 5.5 billion mobile users, representing 70% of the estimated global population for that year (see Figure 1.1). The proliferation of mobile phones, including tablets is growing so fast that more people will own mobile phones (5.4 billion) than those with electricity (5.3 billion), drinking water (3.5 billion) and cars (2.8 billion) in 2020 as it can be appreciated in the following graph [CIS16].

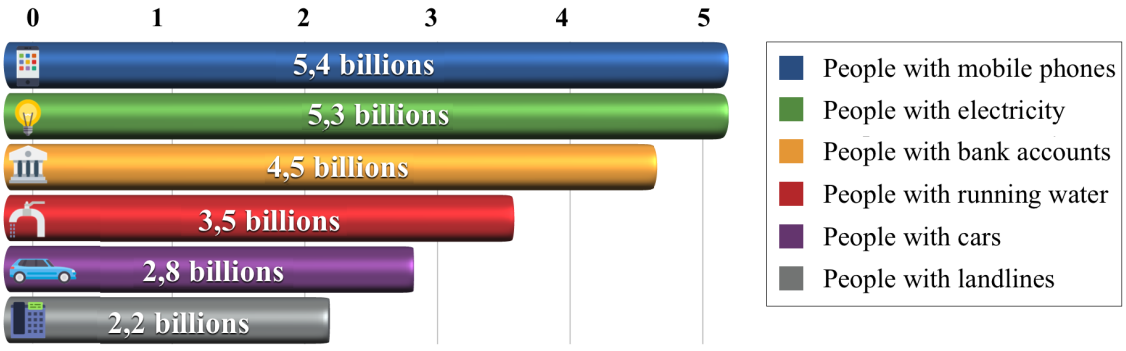


Figure 1.1: Growth of the number of mobile devices until 2020

More than 90% of the mobile devices have an integrated digital camera, which in contrast to conventional digital cameras are carried by their owners all the time to most places they attend and, in many cases, these devices have an Internet connection [AM14]. There are even predictions that DSCs will disappear in favour of the new digital cameras integrated into mobile devices [Bae10], as the quality of these devices increases at an

unstoppable rate. Because of the increase in their storage, processing, usability and portability capacities as well as their low cost, mobile devices are present in diverse activities, places, and events of daily life.

More than 90% of people who have ever taken a picture have done it only with mobile device cameras, a large number of people have and use more than one mobile device, and global statistics show that a typical user on average looks at their mobile 150 times per day, 8 of which is to make use of camera functionality [AM12].

The wide use of mobile device digital cameras is a reality in everyday life. Images and videos generated by mobile devices are seen daily on TV news, diverse applications, email or social networks. Websites like YouTube, Facebook, Instagram or Twitter among others, are placed in the top positions on the list of most visited websites, being a considerable part of its content captured by digital cameras of mobile devices [ALE16]. For these reasons controversies, discussions and rules have been generated about banning the use of mobile devices with digital cameras in places such as schools, government offices, business events, concerts, companies, etc.

Currently, digital images are used as silent witnesses in judicial proceedings, as a crucial piece of crime evidence [AZ06]. In an analogous way in which ballistics tries to relate a gun with its bullets, the forensic analysis of digital images tries to relate a certain image with the digital camera that generated it [WGKM09]. That is why forensic analysis of digital images becomes important, as it could aid in various areas such as the fight against child pornography, prevention of credit card theft, combating piracy, prevention of kidnapping, industrial espionage, etc.

In particular, the study should be focused on mobile devices, since in addition to their extensive use, have intrinsic features that allow better results, not being valid these forensic techniques for digital images generated by other types of devices. In [TNC10] it is described clearly and in detail the need of specific forensic analysis techniques to mobile devices.

1.1 Research Problem

Digital images generated by mobile devices are digital files encoded in a certain format and stored on a media; In most cases, these files contain additional information to the visual content of the image itself, this information is called metadata.

Metadata can be easily removed or modified, whether or not maliciously. Therefore, it may be the case of easily losing track of the device that generated the image, the parameters of geo location, creation date, capture conditions, etc.

Due to the vulnerability metadata, it is necessary to go beyond designing techniques and algorithms that use the image content. In a similar way to metadata, the images can be modified maliciously or not, preventing forensic techniques from being applied. Nonetheless, the techniques and algorithms based on image content are more robust since they require a greater level of knowledge to prevent an analysis perform. These situations could cause problems or uncertainties when images are used as evidence in some process, whether judicial or not, since it is not possible to guarantee image acquisition source

identification or the integrity of the image without performing a forensic analysis.

Regarding to the techniques of acquisition source identification, most of them are based on closed scenarios, these techniques assume that the analyst has a priori knowledge about devices make and model to which the images under analysis may belong. However, there is a need to migrate to open scenarios such as clustering techniques in order to get closer to the reality where any digital image could belong to any mobile device. The counter-forensics techniques study gains value by allowing us to go one step further, and thus to have the possibility of generating new forensic techniques or improving existing ones. Once the problems are described, different counter-forensics and forensics analysis solutions are proposed in this study.

A solution to the image acquisition source identification is proposed; the proposal is based on the use of specific image features. Also, a clustering technique is presented to accomplish the identification in open scenarios; this proposal is based on the combination of hierarchical and flat clustering, and the use of the sensor pattern noise. In the counter-forensics area, a pair of techniques are proposed, the first one to eliminate the possibility of identifying the image source and the second to allow image identity forgery; both techniques are based on wavelet transform and sensor noise.

In the presented solutions development a practical orientation has been considered, in addition, the experiments have been performed with a wide variety of scenarios and mobile devices.

1.2 Motivation

To perform the analysis of a digital image generated by a camera, the features impregnated by the device during image generation are used.

A particular motivation for this work is that most of the research done in recent years about digital image forensics has been focused solely on images generated by traditional cameras DSC.

Considering that today mobile device cameras have practically replaced the DSCs, the need to focus the forensic technique analysis on mobile device images arises. In addition, the need to generate experiments in image source identification and grouping with a greater number of cameras, to represent more realistic scenarios is detected.

In the counter-forensics area, most proposals to eliminate or to forge the acquisition source of an image assume they have access to the victim's camera. Hence the motivation emerged to generate techniques that only require the attacking camera, getting closer to the reality.

1.3 Objectives

This research has five main objectives:

1. Develop a technique to identify digital image acquisition source based on the image features; this technique must be able to identify between different types of devices such as computers, scanners or mobile device digital cameras, as well as between different make and model for the case of the latter;

2. Create a source identification technique for open scenarios based on clustering techniques, focused on mobile devices and without priori known classes or training phase;

3. Design a technique to anonymize the identity of a digital image generated by a mobile device, avoiding being victims of possible attacks;

4. Design a technique to forge the identity of a digital image generated by a mobile device in which is not assumed the access to the victim's digital camera;

5. Integration of the above techniques in the tool for the forensic analysis of mobile devices Theia.

Objectives 1 and 2 are involved in the area of image source identification, objectives 3 and 4 belong to the forensic countermeasures area, and objective 5 is related to these two areas of forensic analysis implementing and integrating the proposed techniques in a software application.

1.4 Summary of the Contributions

The research results achieved in this thesis comprise diverse contributions that have been published in different high impact magazines / conferences. As it is shown in Figure 1.2, these contributions are framed in the area of forensic analysis of mobile devices images.

For the first branch of forensic analysis to be addressed, image source acquisition identification: First, a solution is proposed to the problem of identifying the mobile devices images acquisition source in closed scenarios based on the image features. Secondly, a clustering technique is proposed, based on the combination of hierarchical and flat clustering along with the use of the sensor noise pattern, to accomplish the identification in open scenarios. These two forensic techniques for identifying the mobile devices images acquisition source are organized in two chapters: The algorithm based on image features is presented in Chapter 4 [SOAGRC+13] [RCSOGV15a] [SORCGVHC16] [RCAVSOGV16] and the digital image clustering algorithm is described in Chapter 5 [RCAGSO+13] [RCAGSOGV14] [GVSORC15] [RCSOGV15b] [SOGVAG+15]. These identification techniques have been included in a tool for the forensic analysis of digital images of mobile devices called *Theia* [RCEKSOGV16] [SORCGVAG16].

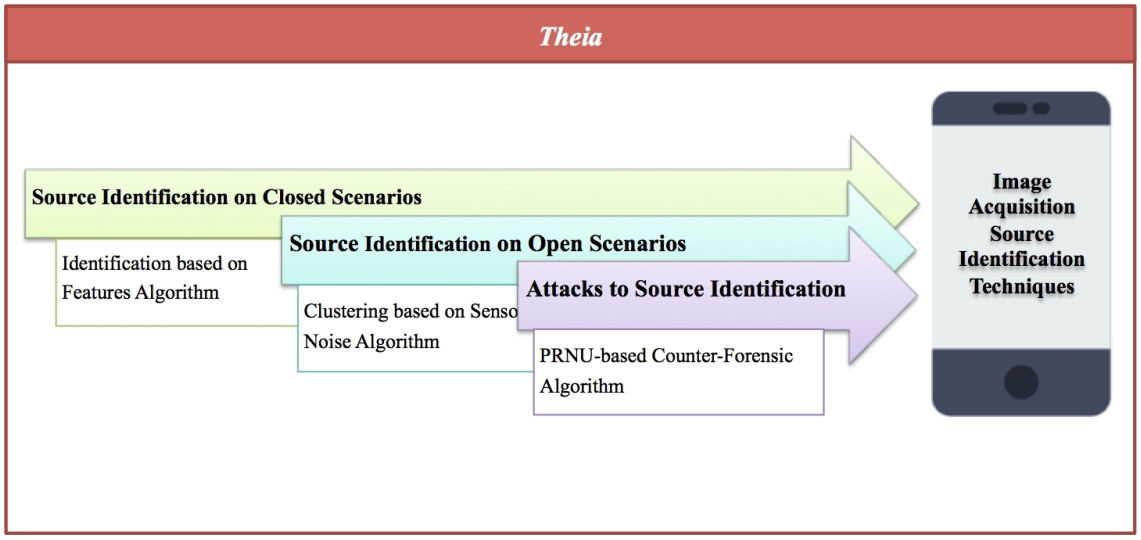


Figure 1.2: Schema of the contributions of this thesis

Finally, in the area of attacks on analysis or anti-forensic techniques, two algorithms for digital images identity destruction and forgery are proposed in Chapter 6 [RCAGSO+13] [RCAGSOGV14] [RCEKSOGV15] [GVSORCHC].

1.5 Outline of the Thesis

This thesis is organized as follows:

Chapter 2 shows some basic aspects concerning mobile device digital images by presenting the image acquisition and creation processes in different types of devices in order to facilitate understanding the next chapters.

In Chapter 3 there is a description of main features that make mobile devices been potential sources of forensic analysis, exposing the different branches of forensic analysis focusing on digital images. Thereafter, the main forensic analysis techniques are explained. The most relevant works on different types of digital images forensic techniques are discussed: acquisition source identification, image clustering [SOAGRC+13]. Then, a study of several attacks that can occur to the digital images forensic analysis is carried out. Finally, a comparative table is presented summarizing the related works shown with in this chapter, highlighting special relevance topics.

The Chapter 4 presents an algorithm based on color features and image quality metrics for the digital image acquisition source identification [SOAGRC+13] [RCSOGV15a] [SORCGVHC16] [RCAVSOGV16]. Initially, the general concepts for understanding the algorithm are presented. Besides, the *Support Vector Machine* (SVM) machines characteristics and configuration used for classification are indicated. The chapter ends with the presentation of the different experiments performed on mobile image banks. The experiments carried out in this chapter have been clearly divided into two distinct groups: acquisition source identification (make and model) and source device type identification (scanner, mobile device or computer).

Chapter 5 develops a mobile device digital image clustering algorithm. Previous concepts are presented to facilitate the algorithm understanding. Next, the proposed algorithm is specified in detail and the results of the different experiments performed are shown [RCAGSO+13] [RCAGSOGV14] [AGRCSO+14] [GVSORC15] [RCSOGV15b] [SOGVAG+15].

Chapter 6 exposes a pair of algorithms based on sensor noise, the first for digital image identity destruction and the second for image identity forgery. The study of these algorithms allows to avoid being victims of possible attacks, as well as to strengthen the techniques of identification [RCAGSO+13] [RCAGSOGV14] [RCEKSOGV15] [GVSORCHC].

Chapter 7 presents the main conclusions of this work, as well as some possible future research lines.

1.6 Audience of this Thesis

The prerequisites for accessing the material in this thesis are not high. Several definitions and concepts are available in the literature, but for the purpose of doing this work independent several definitions and concepts are repeated. Nevertheless, the reader should keep in mind that it is important to have basic theoretical and practical knowledge about digital image forensics, statistics, wavelet transform and computational algorithms.

Some of the problems involved are herein discussed, while it is considered that the reader already has grounds to understand the discussed concepts of some others. The revised bibliography provides the reader with additional information where more details concerning the study conducted by thesis can be found.

Chapter 2

Digital Images

The overall objective of this chapter is to simplify the understanding of the forensic techniques described in the following chapters. The basics of digital image acquisition process in different types of devices, as well as the components involved in this process, are shown. First, the image acquisition process in digital cameras is presented, with particular reference to the *Color Filter Array* (CFA) matrix and to the different sensor types. Subsequently, the image acquisition process in scanners is exposed. The chapter ends with a brief summary of what is presented in it.

2.1 Acquisition Process in Digital Cameras

To get involved in the area of digital images forensic analysis the first step is to know how the devices that generate the image and which one is the creating images process, also known as pipeline. Generally the pipeline presents notable differences between the different types of devices. Within the same device type the pipeline structure is similar and differs in some aspects because of the manufacturer, the quality of camera components or the functionality they offer. Below is the general structure of digital image generating process in cameras and scanners. In the digital cameras section, emphasis is placed on the relevant aspects related to digital cameras from mobile devices. In broad terms a digital camera consists of a system lens, a group of filters, a CFA, an image sensor and a *Digital Image Processor* (DIP) [BSM08]. Many details of the digital camera pipeline belong to each manufacturer and device type and they are considered confidential information, however there is a general structure which is similar for each device type.

In order to generate a digital image, first the lens system collects light from the scene controlling exposure, focus, and image stabilization. Then the light goes through a combination of filters which included at least the infra-red and anti-aliasing filters to ensure maximum image quality. The infrared filter absorbs or reflects light allowing only the visible part of the spectrum passes to the next phase, preventing infrared radiation causes loss of image sharpness. The anti-aliasing filter is responsible for cleaning the signal producing images with smoother contours.

Next, light is focused onto the imaging sensor. There may be mechanisms interacting with the sensor to determine the exposure (aperture size, shutter speed, automatic gain

control) and the lens focal length. The image sensor is an array of light-sensing elements called pixels, which are monochromatic. After light impacts against pixels they generate an analog signal proportional to the intensity of light received, which is converted into a digital signal to be processed by the DIP.

Because the image sensor is monochrome, to capture a color image different sensors are required. Ideally, a sensor for each color. However, due to the cost involved, in practice cameras typically use a single image sensor with a CFA matrix which is placed before the sensor to produce colors. Note that the choice of the CFA can influence the sharpness and the final appearance of the image since there are different CFA patterns.

Once the image processor receives the digital signal generated by the sensor, the noise and other anomalies introduced into the digital signals (artifacts) are eliminated in order to obtain a more visually pleasing image. The most prominent of these processes is the called chromatic interpolation (demosaicing or demosaicking). The purpose of the demosaicing algorithm is to calculate the missing color values because the sensor only provides information on a certain amount of colors (those which the CFA matrix allows passing), this is one of the most complex processes from the computational point of view and the techniques used are usually owned by the camera manufacturer.

Finally, the complete final image is formed and compressed by the DIP. In mobile device cameras usually the algorithm *Joint Photographic Experts Group* (JPEG) is used [Ham92] to save space, storing the image in the device's permanent memory together with the image metadata in *Exchangeable Image File Format* (Exif) [RSYD05]. The digital camera pipeline general structure is shown and summarized in Figure 2.1.

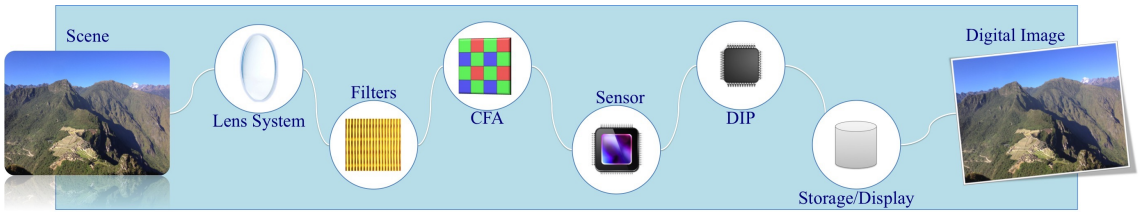


Figure 2.1: Image acquisition process in digital cameras

2.2 Acquisition Process in Digital Scanners

The scanner pipeline has notable differences from the cameras pipeline. Initially a lamp is used to illuminate the document to be digitized. This can be a *Cold Cathode Fluorescent Lamp* (CCFL) which is a xenon lamp, or in the old models of scanners were used normal fluorescent lamps. There is a stabilization bar, which together with an electric motor can move the reading head linearly without having deviations of any kind. The scanner maximum resolution is defined by the number of elements in the linear sensor (horizontal resolution) and the step size of the motor that moves the reading head (vertical resolution). The light reflected by the document when the lamp light hits is transmitted through a set

of mirrors until it reaches the lens.

The lenses are responsible for focusing light to a set of color filters. Each color filter allows light of a single color to pass to a line of sensors, as these sensors like in digital cameras are monochrome. Each element of this line of sensors is generally formed of three individual sensors each with its corresponding color filter. Each sensor therefore captures each of the colors of the additive model (red, green, blue). Most desktop scanners use sensors *Charge Coupled Device* (CCD), although there are also scanners with sensors *Complementary Metal Oxide Semiconductor* (CMOS), *Contact Image Sensors* (CIS) or *Photomultiplier Tubes* (PMT). From the sensors, an analog signal is obtained which is converted into digital in the *Analog Digital Converter* (ADC) module. Once the image has been digitized, the image is then interpolated. The interpolation process basic idea is analogous to digital cameras, but with the peculiarities of the filters and sensors from scanner. That is, by software processing for each pixel a color is formed from the three basic colors captured by each sensor and in certain cases with the adjacent pixels colors. Each manufacturer usually has its own interpolation algorithm. In certain cases, this interpolation process in scanners is also used to increase the perception of the resolution of the image. In other words, new pixels not captured by the sensor are created by processing software, this is known as interpolated resolution. Lastly, as final steps before final digital image creation, software processing is performed to correct possible image defects such as white balance and gamma correction. In Figure 2.2 the general scanner pipeline structure is showed graphically.

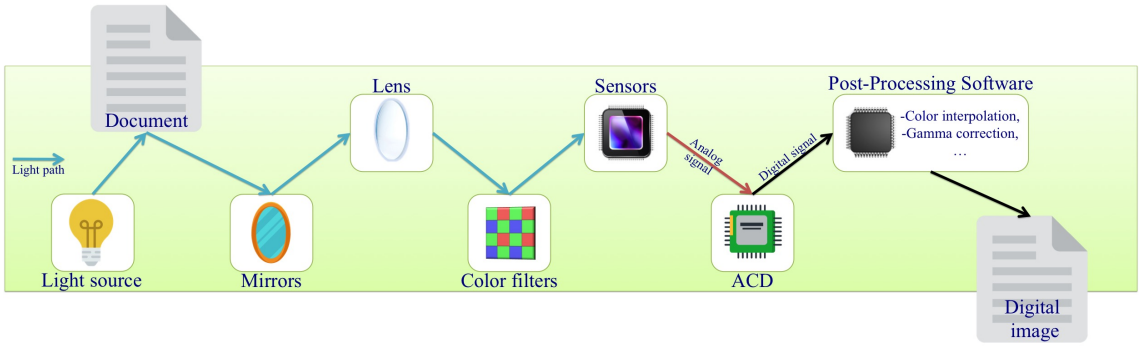


Figure 2.2: Image acquisition pipeline in scanners

2.3 Global Image Structure

2.3.1 Color Filter Array

The CFA matrix is a mosaic of tiny color filters placed on the pixels of the image sensors to capture the color information, is one of the most important parts in the camera pipeline [APS98]. The CFA is located on the monochrome sensor and its function is to acquire the color information of the scene. Each color filter cell passes light according to a range

of wavelengths, so that separate filtered intensities include information about light color. As illustrated in an example in Figure 2.3, the intensity of light passing through each of the cells forms a grayscale image and, depending on the configuration of the CFA filter, is interpreted as a color image (considering that each pixel corresponds to an intensity value).

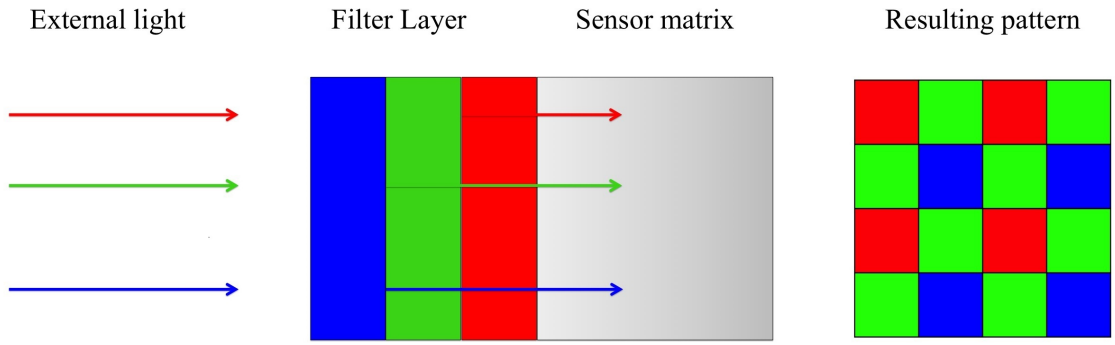


Figure 2.3: CFA matrix

There are values not measured for each of the colors of the filter CFA according to its configuration. The algorithm responsible for calculating these missing values by interpolating the values of neighboring pixels is called demosaicing, it is the most complex process in terms of computation.

The color filters design and the algorithm for performing color interpolation can vary between manufacturers, the latter even when using the same type of CFA matrix. The CFA matrix design used influences the resulting image, both in sharpness, and the edges appearance and small details.

The chromatic interpolation process can generate image anomalies such as aliasing (an effect that produces the unpleasant appearance of staggered lines in the contours of the images), noise and distortions in color. The use of another filter can eliminate the presence of these imperfections in certain image areas at the cost of degrading quality in other [LP05].

There are different CFA patterns as the Bayer model *Green-Red-Green-Blue* (GRGB) shown in Figure 2.3. Other CFA pattern models are the filter *Red-Green-Blue-Emerald* (RGBE), *Cyan-Yellow-Yellow-Magenta* (CYYM), *Cyan-Yellow-Green-Magenta* (CYGM) or the *Red-Green-Blue-White* (RGBW). Figure 2.4 illustrates the color filters mentioned above [Nak05].

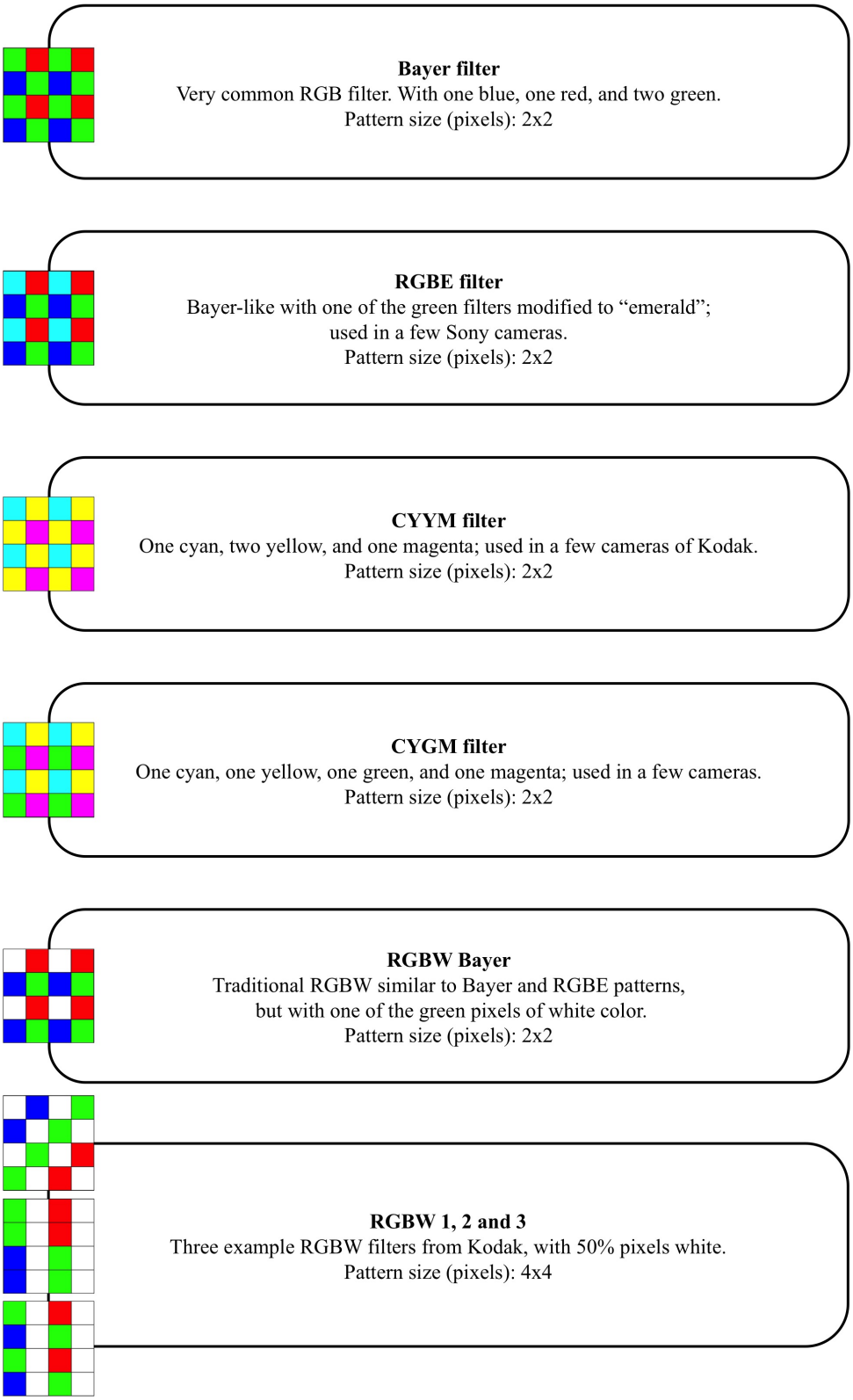


Figure 2.4: Types of color filter arrays

Normally, the cameras use the Bayer pattern model GRGB and the final image generation by the demosaicing. This filter captures for red channel the 25% of pixels, for green 50% and for blue the remaining 25%. This means that for the final image construction 75% of pixels from red channel, 50% from green channel and 75% from blue channel have to be recovered. In Figure 2.5 it can be seen a color capture under this scheme.

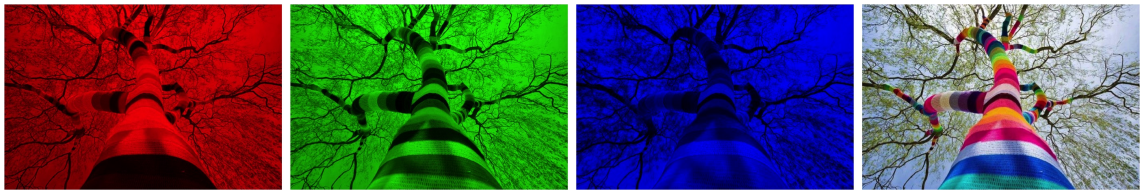


Figure 2.5: Bayer GRGB filter application example on a real image

2.3.2 Image Sensor

The image sensor is the most important part of digital cameras. Generally, it is considered the camera heart. The sensor is an array of light sensitive elements called pixels. Pixels are made of silicon and they capture light converting the photons into electrons using the photoelectric effect. Each pixel is responsible for accumulating the charge induced by the light during a certain time of exposure and then being read and processed. The sensor output signal is proportional to the accumulated charge, depending on the amount of light incident on the pixel and the time of exposure to it.

There are currently two types of sensor technologies used for camera sensors manufacturing: CCD and CMOS. Both types of sensors essentially consist of metal-oxide semiconductors *Metal Oxide Semiconductor* (MOS) distributed in matrix form. They work in a similar way, although the key difference is in the way in which pixels are scanned and the way in which the reading of the charges is carried out.

2.3.2.1 Charge Coupled Device Sensor

In CCD sensors each of the matrix cell loads are transformed into voltages and an analog signal is delivered as output in order to later digitized it by the camera. The structure of this type of sensors is very simple, but has the disadvantage of having an additional chip to process the sensor's output information; this causes the manufacture of devices to be more costly and the sensors to be bigger. The pixels in a CCD array capture light simultaneously, which promotes a more uniform output.

Unlike CMOS sensors that support reading the pixel array in a random way, in CCD sensors all pixels begin and end load integration at the same time. This leads to a more uniform output (a expected result from a pixel subjected to the same level of excitation of the others without any noticeable changes in the signal obtained). This type of exposure

is known as global shutter. It is possible to add circuits to CMOS sensors making them obtain a similar result, however CCD sensors still being over them.

2.3.2.2 Metal Oxide Semiconductor Sensor

CMOS sensors have an independent and active pixels design. They are called active pixels because the digitization is done internally by themselves in some transistors that offer better processing speed, eliminating the need for an external chip that performs this function, which reduces the cost and size of the equipment. The independence characteristic refers to the flexibility that this type of sensors offers for reading the pixel array, since it is possible to access each cell by its row and column position. Generally, the reading of the matrix is performed as a progressive scan. This scheme is known as rolling shutter (It is not necessary to read the entire matrix in a single time as in the case of sensors CCD). In addition, since the CMOS are made up of independent cells, they do not have the blooming effect. This effect occurs when a pixel is saturated by the light that hits it and then begins to saturate those around it.

A further advantage is that CMOS sensors are more sensitive to light and behave better in low light conditions. Additionally, because the signal amplifiers are within the same cell, no extra power consumption is generated unlike the sensors CCD.

Signals stored by the CCD/CMOS sensor are then converted into a digital signal and transmitted to the image processor, once the image processor receives the digital signal it eliminates noise and other introduced anomalies. Some other processes applied to the signal are color interpolation, gamma correction, and color correction.

In their beginning CMOS sensors were not considered as good as the CCD sensors. Nevertheless, the CCD technology has reached its limit and it is now when CMOS is being developed and its weaknesses are being overcome; so much so that there are sensors called Smart CMOS sensors for the purpose of improving the deficiencies of conventional CCD sensors and CMOS sensors. Most of the *Digital Still Camera*s (DSCs) use CCD sensors and on mobile devices the use of sensors CMOS is more common.

2.3.2.3 Sensor Imperfections and Noise

Imperfections are artifacts that remain constant from image to image, while noise is considered a random artifact, much like the static on a television set. Sensor noise will not survive frame averaging, while sensor imperfections will. However generally in the algorithms described below sometimes the sensor imperfections are referred as sensor noise, pointing to the characteristics that remain constant.

Image acquisition for any given imaging device is complex and varies depending on the equipment and manufacturer. However, there are similar types of noises that are inherent in each device, both random and systematic. Shot and quantization noises are erratic and do not have consistent or predictable patterns. Shot noise is a result of the non-continuous flow of electrical current and is the sum of discrete pulses in time for each pixel.

The longer a sensor is active, i.e. longer shutter speeds, or the more sensitive the sensor is, i.e. low light conditions, a higher number of random electron noise will be

recorded by the image sensor and recorded along with the scene. This type of noise is temperature dependent, meaning that higher temperature conditions will cause higher electron movement in the circuitry than lower temperatures. Quantization noise is caused by the process of converting light from an infinite amount of intensity values into a digital medium that has a finite amount of intensity levels. While this process introduces small distortions into the image, finer detail with larger bit depth can minimize this error.

Most research on image forensic analysis source acquisition identification focuses on traditional digital cameras or DSCs; most of these techniques are not valid for mobile device images. The main reason is that most of the techniques are based on directly or indirectly use of sensor features or in the lens of the digital camera. Regarding the sensor, it is the component that is responsible for capturing the light and generate a digital signal according to its intensity.

The CCD sensors are by far better than the CMOS ones in regard to the dynamic range (coefficient between the pixels saturation and the threshold below which they do not capture signal), being as they are less sensitive they tolerate the ends of light better. Also, CCD sensors are superior to CMOS in terms of image noise, since the signal processing is performed on an external chip that can be optimized for the performance of this function. In contrast, CMOS sensors perform signal processing within the same sensor leaving less space to place light-collecting photo-diodes.

Early CMOS sensors were somewhat worse than CCDs, but nowadays this has been practically corrected. The CCD technology has reached its limit and nowadays the CMOS technology is developing and gradually overcoming their shortcomings. Most of DSCs use CCD sensors, in mobile devices is more common to use sensors CMOS. Even day by day, reducing the quality differences between CCD and CMOS sensors, in the great majority of cases DSCs sensors notably exceed in quality to sensors in mobile devices digital cameras, and this is a strong reason to require specific techniques for image source acquisition source. Likewise to the case of sensor, mobile devices digital camera lenses, in general, are lower of quality than DSCs lenses.

The main components of image noise are the *Fixed Pattern Noise* (FPN) and the *Photo Response Non-Uniformity* (PRNU). There are several sources of imperfections and noise introduced at different stages of the creating pipeline of an image in a digital camera. Even if a uniform and fully lighted picture is taken it is possible to see small changes in the intensity between pixels. This is due to the shot noise is random and, in large part, the pattern noise is deterministic and is kept approximately equal if several pictures of the same scene are taken.

The noise pattern of an image refers to any spatial pattern that does not change from one image to another. It is composed of the spatial noise which is independent of the signal (FPN) and of the spatial noise due to the difference in the response of each pixel to the incident signal (PRNU).

Noise FPN is generated by the dark current and it also depends on exposure and temperature. Since the FPN is an independent additive noise, some cameras automatically removed by subtracting a dark frame to generated images. The noise pattern structure is shown in Figure 2.6.

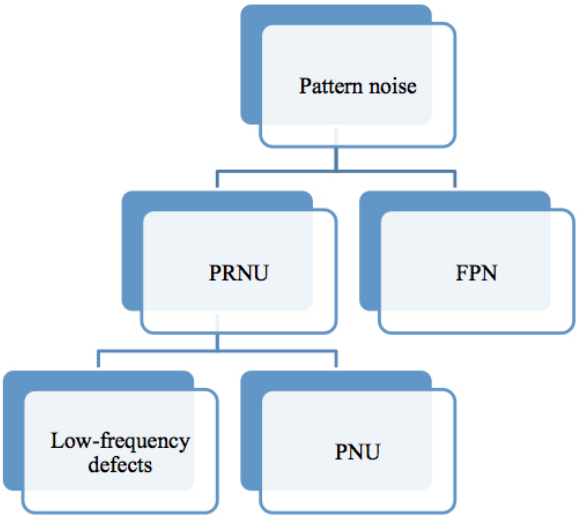


Figure 2.6: Sensor pattern noise

2.3.3 Photo Response Non-Uniformity

Noise PRNU is the dominant part of the sensor pattern noise of an image and it is a multiplicative noise dependent. Noise PRNU is mainly formed by noise *Pixel Non-Uniformity* (PNU) and by the low frequency defects as zoom settings and light refraction in the dust particles and lenses. Noise PNU is the light sensitivity difference between pixels of the sensor array. It is generated by the lack of homogeneity of the silicon wafers and by the imperfections during the sensor manufacturing process. Due to the nature and origin, it is very unlikely that even the sensors from the same wafer have PNU correlated patterns. This noise is not affected by ambient temperature nor by humidity. Noise PNU is usually more common, complex and significant in CMOS sensors, due to the complexity of pixel array circuitry.

2.4 Summary

The aim of this chapter has been to introduce the digital images acquiring process. It has begun with the explanation of image pipeline in digital cameras. Special emphasis has been made on the pipeline elements that could be studied for future forensics techniques of acquisition source identification. An explanation of the matrix CFAs, is done as it is considered that the specification of this along with the color interpolation algorithm used to produce the most significant differences between different camera models. In the same way, a description of the different sensors types has been made, in order to clarify the differences in this element between DSCs and digital cameras on mobile devices. Finally, we have shown the scanners pipeline, which, despite having common elements of a digital camera, differs markedly.

Chapter 3

Forensics in Digital Images

This chapter presents the state of the art by classifying the related works in the following groups: techniques for identifying the image acquisition source, image clustering techniques and attacks on digital image forensics. It begins with the classification of image forensic analysis and continues with the exposition of related works concerning the digital image acquisition source identification. Subsequently, the work related to image clustering is presented. Afterwards, a section shows the counter-forensics classification and the works regarding the attacks on digital image forensics. The chapter ends with a brief summary of what is presented in it. It should be noted that although this work is focused on mobile devices, there are also references to techniques on images from all types of devices since their knowledge can be valid for the application or adaptation to images on mobile devices.

3.1 Image Source Acquisition Identification Techniques

In this section we describe the main techniques of digital image forensics for identifying the source of image acquisition and the main work of the analysis. Other compendiums of techniques may be shown in [SWL09] [RSBG11] [SOAGRC+13].

As stated in [CFGL08], the tasks of digital forensics can be broadly divided into the following categories:

- **Source Classification**: Its objective is to classify images according to their origin, such as scanners or digital cameras.
- **Device Identification**: It aims to prove that a given image was obtained by a specific device (make and model).
- **Device Clustering**: Given a set of images, it tries to find out which images were obtained using the same camera.
- **Processing History Recovery**: Its purpose is to recover the processing chain applied to the image. Here, we are interested in non-malicious processing, e.g., cutouts, filtering, contrast adjustments, etc.

- **Integrity Verification or Forgery Detection**: It is a procedure aimed at discovering malicious processing, examples of which include the removal or adding of objects.

The graphical representation of this classification is summarized in Figure 3.1.

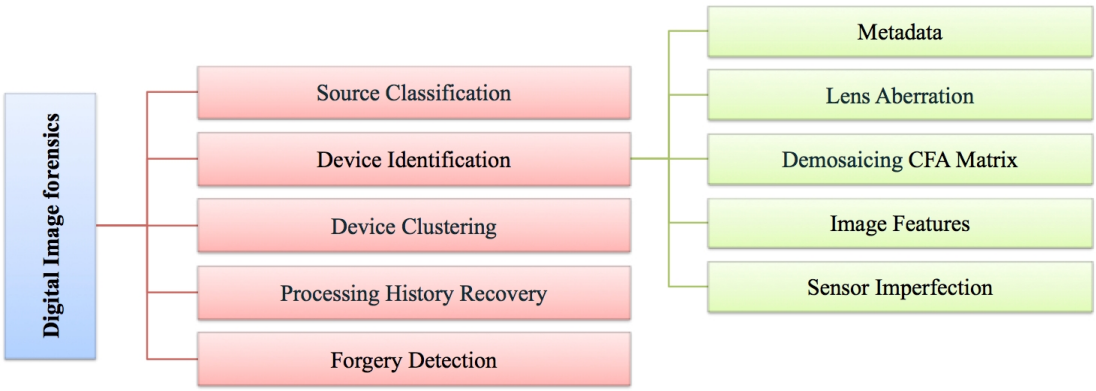


Figure 3.1: Classification of digital image forensic analysis

The success of these techniques depends on the assumption that all the images acquired by the same device have intrinsic features. The features which are used to identify the make and model of a digital camera are derived from the differences between the techniques of image processing technologies and the components which are used. The biggest problem with this approach is that different models of digital cameras use components of a small number of manufacturers, and the algorithms used are also very similar between models of the same brand. According to Van Lanh et al. [VLCEK07] for this purpose four groups of techniques can be established depending on their base: lens system aberrations, CFA interpolation, image characteristics, and sensor imperfections. In addition to the above there is another group of techniques based on metadata.

3.1.1 Techniques Based on Metadata

Techniques based on the image metadata use the information stored in the camera about the conditions of image capture in order to find information and image classification. Digital cameras have a powerful source of information which is the embedded metadata in digital images files. Metadata, or "data about data" store information related to the conditions of image capture, as the date and time of acquisition, flash presence or absence, object distance, exposure time, shutter opening and *Global Positioning System* (GPS) information among others. In other words, it provides relevant information to supplement the main content of a digital document.

The Exif specification [Sta10] is the most common container of metadata in digital cameras [Bae10]. The Exif specification includes hundreds of labels, among which are

make and *model*, although it should be noted that the specification does not make their existence in the image files compulsory.

These techniques are the simplest. There are plenty of studies focused on the different types of metadata, both for finding information and for image classification [Pla00] [BL05] [Tes05] [RCC+08]. Metadata can also be used as input or aid for other forensic techniques. For instance, in the application of content-based image techniques, Exif metadata can provide a large number and variety of technical information, which may allow an increase in the success rates or improve the results of the application of certain forensic algorithms [BL04] [JLLC07] [FKCS11].

However, these techniques depend largely in the metadata inserted by manufacturers when the image is created and the correction. Sandoval Orozco et al. [SOAGGVHC14] make an in depth study on this topic. Moreover, this method is the most vulnerable to malicious alterations.

3.1.2 Techniques Based on Lens Aberration

Techniques based on lens aberration study several types of aberrations introduced by the lens system during the image generation process.

During the image generation process the lens system can introduce some aberrations. There are several types of aberrations: spherical, coma or comatic aberration, astigmatism, field curvature, radial distortion and chromatic aberration. The radial distortion is the one with the most impact over pictures, especially in cameras with cheap wide angle lenses. Most digital cameras use this type of lens for economic reasons.

Choi et al. [Cho06] propose the lens radial distortion as the best technique for source identification. The authors conclude that each camera model expresses a unique pattern of radial distortion which helps to uniquely identify it. They experimented with three different cameras and obtained accuracy between 87% and 91% in identifying the source camera.

Lanh Tran Van et al. [VEK07] propose lateral chromatic aberration as a technique for identifying source camera. The authors performed experiments using little sets of cameras with non-modified images or modified images with random crops regions. In the experiment in which three cameras of different brands were used 86.67% accuracy in identifying the source was obtained. It was concluded that this technique is not suitable for identifying the source of different camera models from the same brand.

3.1.3 Techniques Based on CFA Interpolation

Some authors consider that the choice of the CFA matrix and the specification of color interpolation algorithms produce some of the most significant differences between different camera models [BSM06] [CAS+06] [LH06] [BSM08].

Commercial cameras have a single sensor instead of multiple sensors for each component color. In essence, the color interpolation introduces a specific type of correlation between the color values of image pixels. The specific form of these

dependencies can be extracted from the images to differentiate the color interpolation algorithms and determine make and model of the camera that generated an image.

Long et al. [LH06] use correlations between pixels for the source identification. Neuronal networks are used for classification. The method was tested with cartoon images from four cameras and the success rate obtained was between 95% and 100%, with an average accuracy of 98.25%. Tests for modified images were also made with 80% success rate for a 80% JPEG compression. Since the cameras from the same manufacturer use the same color interpolation algorithm, this approach is not efficient at differentiating between different models from the same maker. Also, as shown in the experiments, good results are not obtained when the images have been modified or when they have high compression level.

Celiktutan et al. [CAS+06] use a set of binary similarity measures as metrics to estimate the similarity between image bit planes. This work uses a set of 108 binary similarity measures. The success rate of their experiments was between 81% and 98% to classify three cameras and decreased to 62% to identify between nine cameras. The results of the method depends on the number of cameras used in the experiments.

Bayram et al. [BSM08] present an algorithm for identifying and classifying color interpolation operations. This proposal is based on two methods to perform the classification process: first using an algorithm to analyze the correlation of each pixel value with its neighbors' values, and secondly an analysis of the differences between pixels independently. Different experiments with different numbers of cameras and image types were performed. The accuracy for the source camera identification had between 84.8% and 92.56% of average success rate.

Cao and Kot [CK10] present a technique for source identification based on the information of the CFA matrix interpolation process and a comparison with other techniques. This technique has three new sets of demosaicing features: weights, *Error Cumulants* (EC) and *Normalized Group Sizes* (NGS). Since the number of features is very high a process (*Eigenfeature Regularization* (ERE)) is performed to decrease the number of it. Different experiments were performed using classifiers *First Nearest Neighbor* (1NN) and *Probabilistic Support Vector Machine* (PSVM). The results using 15 cameras from four different manufacturers and 11 different models (there are cameras of the same brand and model), with a reduction to 20 features and PSVM classifier, obtained an average success rate of 99.4% for the brand identification and 94.8% for model identification.

Ho et al. [HAZG10] propose four algorithms which are based on aspects of inter-channel correlation. These algorithms calculate variance maps (v-maps) and classify using 1NN. The experiments image source identification uses four cameras for three different manufacturers and 50 images of each camera (25 for training and 25 test). The results show an average accuracy of 94.5% and The authors conclude that the inter-channel correlation provides a complementary approach to previous studies which dealing with correlations between pixels introduced in the demosaicing process.

3.1.4 Techniques Based on Image Features

Techniques based on image features use a set of features extracted from the content of the image to identify the source. These features are divided into three groups: color features, *Image Quality Metrics* (IQM) and wavelet domain statistics.

Tsai et al. [TLL07] propose a method to identify the source using the following features: color features, image quality metrics and frequency domain. The study adopted the wavelet transforms as a method to calculate the wavelet domain statistics and use a SVM for classification. In experiments digital cameras and mobile devices were used. The results obtained in different experiments show results between 61.7% and 99.72% accuracy. Tsai et al. [MSGW08] extend the source identification to different devices such as mobiles, phones, digital cameras, scanners and computers. In this proposal they base it on the differences in the image acquisition process to create two features groups: color interpolation coefficients and noise features. In the experiments they use five smartphone models, five digital camera models and four scanner models to identify the source type. Their experiments showed an overall result of 93.75% accuracy. Identifying the maker and model of five mobile phone models resulted in an accuracy of 97.7%.

Wang et al. [WGKM09] propose a method for source camera identification based on the extraction and classification of wavelet statistical features. Finally 216 first-order wavelet features and 135 second order co-occurrence features is obtained. The most representative features are selected using an *Sequential Floating Forward Selection* (SFFS) algorithm and they are classified using a SVM. Identification success average of 98% the set of all cameras and an average success rate of 96.9% for the three cameras of the same model is achieved.

Hu et al. [HLZ10] perform experiments with common imaging features to identify the source: wavelet, color, IQM, statistical features of difference images and statistical features of prediction errors. In the experiments, different combinations of different types of features are used and a SVM for classification of different devices. Ten different cameras from four different makers with 300 images from each camera (150 for training and 150 for testing) and a resolution of 1024×1024 is used. Using all the features a score of 92% success rate is obtained. Moreover experiments were performed to check the robustness against three of the most common alterations in digital images: JPEG compression, cropping and scaling. The final conclusions of this work are that some of the feature sets provide good success rates for intact images, but not for images with modifications. It also shows that different types of manipulations have different effects on success rates of different feature sets.

Ozparlak et al. [OA11] propose a technique for image source identification using ridgelets and contourlets subbands statistical models. After the feature extraction a SFFS algorithm is used for feature election and a SVM for classification. The method based on 216 wavelet features is considered useful only for the representation of a dimension, the approach based on ridgelets uses 48 features, and the approach based on contourlets includes a total of 768 features. In experiments with three cameras from different makers success rates are between 99.5% and 99.8%. The contourlets and ridgelets are not only effective in differentiating between camera models, but also to differentiate between natural

images or those produced by computer, or to differentiate between images from scanners of the same maker. However the authors believe that improvements could be implemented experimenting with different selection algorithms.

Liu et al. [LLC+12] propose a method using the marginal density *Discrete Cosine Transform* (DCT) coefficients in low-frequency coordinates and neighboring joint density features from the DCT domain. Furthermore, hierarchical clustering and SVM is used to detect the source of acquisition of the images. In experiments with images from five smartphone models of four makers an accuracy of between 86.36% and 99.91% was obtained, achieving the best results with a linear SVM kernel.

Sandoval et al. [SOAGRC+14] propose the mixture of two techniques (Sensor Imperfections and Wavelet Transforms) to get the source identification of images generated with mobile devices. This method extracts the sensor noise patterns of images, and then, a set of 25 features are obtained (16 first-order and higher-order features and 9 features by applying *Quadrature Mirror Filter*s (QMFs)). In the experiments in which 10 cameras of 6 different brands were used, an accuracy of between 89.46% and 94.22% in identifying the source was obtained.

3.1.5 Techniques Based on Sensor Imperfections

Techniques based on Sensor imperfections study the fingerprints which can leave sensor defects on pictures. These techniques are divided into two branches: pixel defects and *Sensor Pattern Noise* (SPN). In the first pixel defects, hot pixels, dead pixels, row or column defects and group defects are studied. In the second pattern noise by averaging multiple noise residuals obtained by any noise removal filter is constructed. The presence of the pattern is determined using a classification method as correlation or SVM.

Geradts et al. [GBK+01] study pixel defects of CCD, sensors, focusing on different features to analyze images and then identify their source: CCD sensor defects, the file format used, noise introduced in the image and watermarking introduced by makers. Among the CCD sensor defects are considered hot spots, dead pixels, group defects, and row or column defects. Results indicate that each camera has a different defect pattern. Nevertheless, it is also noted that the number of pixel defects for images from the same camera is different and varies greatly depending in the image content. Likewise, it was revealed that the number of defects varies with temperature. Finally, the study found that high quality CCD cameras do not have this kind of problem. When considering only defective CCD sensors this study is not applicable to the analysis of images generated by mobile devices.

Lukas et al. [LFG06] analyze the sensor pattern noise from a set of cameras, which functions as a fingerprint allowing the unique identification of each camera. This pattern noise is obtained averaging the sensor noise extracted from different images with a noise removal filter. To identify the camera from a given image, the reference pattern is considered as a watermark in the image and its presence is established by a correlation detector. The study was done with approximately 320 images from 9 cameras (2 are exactly the same model) and good results were obtained. It is noted that this success rate is because in the experiments the authors used the same set of images to

calculate the reference pattern and correlations. It is also shown that This method is affected by processing algorithms such as image JPEG compression and gamma correction. According to Van Lanh et al. [VLCEK07] the results for pictures with different sizes were unsatisfactory. Also in this technique, images whose reference pattern is extracted must have the same size as the test images.

Costa et al. [CESR12] propose an approach to source camera identification in open set scenarios, where unlike closed scenarios it is not assumed to have access to all the possible image source cameras. This proposal includes three phases: definition of regions of interest, determining the characteristics and source camera identification. Different regions of the images can contain different information about the fingerprint of the source camera. Besides, this approach in contrast to others considers 9 different *Region of Interest*s (ROIs), not only the central region of the image. Using these ROIs it is possible to work with different resolution. For determining the features the SPN for each of the R, G, B and Y (luminance) channels is calculated, generating a total of 36 representative features for each image.

Then, the features of images taken by the camera under investigation are labeled as positive class and features from images made by other cameras as negative classes. After the SVM training phase in which the hyper-plane that separates the positive and negative classes is estimated. Later, the unknown classes of open stage are taking into account, moving the generated hyper-plane toward the positive classes or to the negative classes. By moving the hyper-plane the margin can change to determine if an image belongs to one class or another. This process is called modeling decision boundaries. In the experiments a set of 25 digital cameras of 9 manufacturers, 150 images of each camera in JPEG format with different light configuration, zoom and flash are used. The results of the experiments showed a success rate of 94.49%, of 96.77% and 98.10%, using open sets with 2/25, 5/25 and 15/25 cameras, respectively, defining open x/y as the set of y cameras where x cameras are known and used for training and $y - x$ are unknown cameras, whose images and the images of the known cameras are used in test stage.

Costa et al. [dOCSE+14] extended this approach, where in addition to presenting other techniques and algorithms, new experiments are performed. In experiments 13210 images of 400 cameras were used (they only have physical access to 25 cameras, the rest are images downloaded from Flickr) and they obtained better success rates of 96.56%, 97.34%, 96.80% and 97.18%, using open sets with 2/25, 5/25, 10 /25 and 15/25 cameras respectively.

3.1.6 Summary Table of Image Source Identification Techniques

Table 3.1 shows a summary of the findings described above. The no detailed information in articles has been filled with *Non Detailed* (ND). We must take into account that in most of the above articles various experiments with different numbers of cameras and images are performed. In the column "Number of models/makers" the total models and manufacturers used for all experiments are accounted, which does not imply that in all experiments all models from all manufacturers are used. The "Applied to mobile devices" column indicates that at least one of the models used in the experiments is a mobile device. The column "Applied to different models of the same brand" indicates that at least in one experiment cameras from the same manufacturer were used. In each experiment an average success rate is obtained in the source identification, the "Minimum and maximum success rate" column shows the minimum and maximum values for the different experiments (in the case that there is only one value is because this article only has one experiment).

Table 3.1: Evaluation of camera identification techniques

Technique	Proposal	Classification Method	Number of Models / Makers	Image Formats	Resolution	Applied to Mobile Device	Applied to Different Models of the Same Brand	Minimum and maximum success rate
Lens Aberration	[Cho06]	SVM	3/3	JPEG	Different	ND	No	87.38% - 91.53%
	[VEK07]	SVM	3/3	JPEG	Different	No	Yes	72.75% - 92.22%
CFA Interpolation	[LH06]	Neuronal Network	4/4	Uncompressed	ND	No	No	98.25%
	[CAS+06]	SVM Linear and Non-linear RBF	9/3	ND	Different	Yes	Yes	62.3% - 98.7%
	[BSM08]	SVM Linear and Non-linear RBF	5/5	JPEG	Different	No	Yes	84.8% - 88%
	[CK10]	PSVM and 1NN	4/11	JPEG	Different	Yes	Yes	94.8%-99.4%
	[HAZG10]	1NN	4/3	JPEG	ND	No	Yes	94.5%
Image Features	[TLL07]	SVM Linear	2/7	JPEG	1600×1200	Yes	Yes	61.7% - 99.72%
	[MSGW08]	SVM Linear	5/5	JPEG	ND	Yes	No	97.7%
	[WGKM09]	SVM Non-linear RBF	6/4	JPEG	ND	No	Yes	98%
	[HLZ10]	SVM Non-linear RBF	10/4	JPEG	1024×1024	No	Yes	47% - 92%
	[OA11]	SVM Non-linear RBF	3/3	ND	ND	No	No	93.33 - 99.7%
	[LLC+12]	SVM Linear and Non-linear RBF	5/4	JPEG	Different	Yes	Yes	86.36% - 99.91%
	[SOAGRC+14]	SVM	10/6	JPEG	ND	Yes	Yes	89.45%
Sensor Imperfections	[GBK+01]	ND	2/2	ND	640×480	No	No	ND
	[LFG06]	ND	5/9	Different	Different	No	Yes	ND
	[CESR12]	SVM Non-linear RBF	25/9	JPEG	Different	Yes	Yes	94.49% - 98.10%
	[dOCSE+14]	SVM Non-linear RBF	25/9	JPEG	Different	Yes	Yes	96.56% - 97.34%

3.2 Clustering Techniques of Digital Images

Once you have the features to be used for classification of images we will focus on issues relating to the classification by clustering. The analysis of clusters, or clustering, aims to group a collection of objects into representative classes called clusters, without a priori information, in such a way that the objects belonging to each cluster keep a greater similarity to objects from other clusters.

Image grouping can be performed using supervised or unsupervised learning techniques. In the first case it is essential to know the device information a priori, i.e., it is clearly identified with the classification in closed scenarios which requires a training stage with the features extracted from the images and a second classification stage in accordance with the previous result. However, in a real case it may be difficult to have the camera in question or a set of photographs taken by it to carry out training, hence the need for unsupervised learning techniques, which directly correspond to open scenarios.

Traditional clustering has been known to be an unsupervised learning technique; however, there are some cases of supervised clustering where it is possible to apply an anterior or posterior approach to improve the grouping itself. This is to prevent that elements of different classes are in the same cluster, which requires having a priori knowledge of the data set. This issue is addressed in [EZZ04], although it is worth mentioning that this article is focused on the use of unsupervised techniques. In order to determine the similarity between objects belonging to the same cluster, there are distance measures such as Euclidean distance, Manhattan distance, and Chebychev distance, among others. Alternatively, it is possible to use similarity functions $S(X_i, X_j)$ which compare two vectors X_i and X_j symmetrically, i.e., $S(X_i, X_j) = S(X_j, X_i)$. These functions reach their highest values as X_i and X_j are more similar.

One of the most commonly used measures in image source identification is normalized correlation [Blo08, Fri09, Li10b, CAPI10] defined as:

$$corr\,(X_i, X_j) = \frac{\left(X_i - \overline{X_i}\right) \odot \left(X_j - \overline{X_j}\right)}{\left\|X_i - \overline{X_i}\right\| \cdot \left\|X_j - \overline{X_j}\right\|} \tag{3.1}$$

where $\overline{X_i}$ and $\overline{X_j}$ represent the mean vector, $X_i \odot X_j$ is the scalar product of two vectors and $\|X_i\|$ is the L_2 norm of X_i.

According to the clustering algorithms classification proposed in [Rok10], we find the hierarchical methods whose purpose is to achieve a structure called dendogram (See Figure 3.2) which represents the grouping of objects according to their levels of similarity.

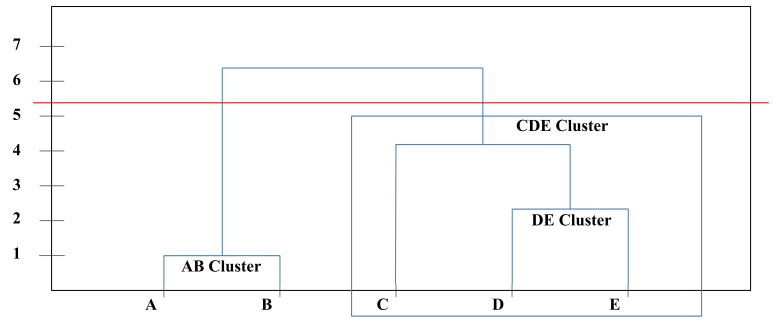


Figure 3.2: Example of dendogram

This grouping can be done in different ways: agglomerative or decisive. Agglomerative grouping initially considers each object as a separate class until iteratively grouping all the objects in a single class. Divisive clustering is based on the idea of starting from a single class until managing to separate all objects into individual classes. There are also partitioning algorithms, wherein starting a partition, the algorithm takes care of moving objects from one cluster to another to minimize certain error criterion. Within this category, the most famous method is k-means; however, most of these methods require knowing in advance the number of clusters, which is why they are not widely used in forensic image analysis.

Finally, there are other clustering algorithms such as: [Zah71] which produces clusters by means of graphs, [BR93] based on the density where the points within a cluster are given by a certain probability function, clusters based on models such as decision trees [Fis87] or neural networks [VA00] and clustering with soft-computing methods such as fuzzy clustering [HKKR99], evolutionary clustering methods and simulated annealing clustering [SA91].

[XW05] shows a comprehensive review of the different types of clustering algorithms, as well as an extensive review of approaches used on this subject in the state of the art. Among other aspects, it is concluded that there is not a universal clustering algorithm to solve any kind of problem and therefore approaches to clustering for each field or situation may be completely different. It also highlights the importance of the stage of selection and extraction of the characteristics of the elements to be classified.

There are previous works on image grouping by unsupervised methods; all of them consider SPN as the most reliable criterion for representing a device's digital footprint, hence the SPN is used specifically as a footprint and normalized correlation as a similarity measure to achieve image grouping by device.

Once seeing an overview of the different types of clustering algorithms, then will present some of the related works that deal with the clustering of images using the SPN.

[LLHC10] uses a classification technique with unsupervised learning where grouping is achieved by graph maximization. Clustering is performed from not-oriented graph with weights, starting with an affinity matrix where the connection weights between vertices is the correlation value between each SPN, starting with a random node. In each iteration, the remaining nodes are connected and the nodes closest to the central one are chosen,

obtaining a new affinity matrix in each step; the algorithm stops when the number of closest nodes is less than a k parameter. Subsequently, the graph is partitioned to the point where similarity in a set is maximum and minimum with respect to other sets.

In [Li10b] clusters are performed using Markov random fields. A clustering algorithm based on matrix containing all the correlations between the SPN of several cameras is proposed. In each iteration the algorithm groups within classes the most similar SPNs making use of the local features of Markov random fields and assigns a new class label to each SPN maximizing a probability function, the criterion to stop the algorithm is satisfied when there are no label changes after a certain number of iterations.

The algorithm proposed in [CAPI10] and on which this research is based uses hierarchical clustering to group images. Prior to the clustering algorithm, the authors apply a function for sensor noise improvement, which strengthens the lower components and attenuates the high components in the wavelet domain in order to remove the scene details in it. With a similarity matrix containing all the correlations between different SPNs and taking as a starting point each image as a single cluster, the clustering algorithm groups the two clusters with the highest correlation value forming a single cluster and updates the matrix with a new row and column that replace the rows and columns of the grouped clusters. The link criterion chosen to mix two clusters was average linkage. In each iteration of the algorithm, cluster status at that time is stored on a partition and the global silhouette coefficient is calculated. At the end of the algorithm the partition whose silhouette coefficient value is the lowest is chosen, the number of clusters at that point should correspond to the number of devices that exist initially, as well as the content of each cluster to the SPN for each device. The authors carry out a training stage with the described algorithm and a classification stage for the remaining images, for this it is sufficient to obtain the average of the SPNs for each cluster and compare them against the remaining images, the image will be classified within the cluster whose correlation is highest.

3.3 Attacks on Digital Image Forensics

In contrast to the prominent role of digital images in our society today, research in the field of image authenticity is still in a very preliminary stage. Most publications in this emerging field still lack rigorous and robust discussions against strategic counterfeiters, who anticipate and try to fool the use of forensic techniques [GKWB07].

The area in charge of studying attacks on imaging forensic techniques is known as counter-forensics. Attacks on digital image forensic algorithms are aimed at systematically confusing or misleading the procedures for identifying the source of an image or detecting malicious image manipulations. These attacks could have one of the following goals: camouflage malicious post-processing of images or manipulating the image source identification.

With respect to the aim of manipulating the image source identification, as well as for the process of source identification, the sensor noise extracted from images is generally used. A logic counter for this technique consists in removing all the sensor noise. Taking

a step further, one could also think of the possibility of removing the sensor noise and replacing it with the sensor noise belonging to another camera. Therefore this goal can be divided into two branches: destruction of image identity or forgery of image identity (destruction or forgery). An outline of this classification is shown in Figure 3.3.

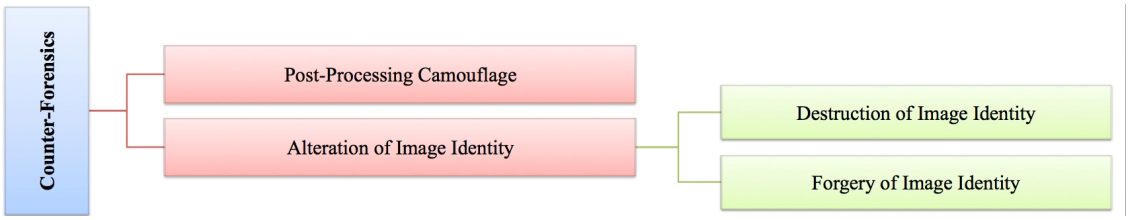


Figure 3.3: Classification of counter-forensics

Below, some of the general concepts of each of these three goals of counter-forensics will be treated, as well as some proposed solutions.

3.3.1 The Post-Processing Camouflage

These techniques are designed to hide that different processes have been applied to an image. This is achieved by analyzing the traits left by processes on the image during their application, in order to counter them.

In [PF05] the dependencies introduced during the resizing or rotating process of digital images are examined in detail.

In [LF03] authors study the statistical coefficients of JPEG to detect recompression.

In [CSS07] the phase congruence is analyzed to detect image composition done by cutting and pasting parts of different images.

In [GKWB07] a proposal to hide resampling, which is the process of resizing images with interpolation, and extremely common in operations like scaling and rotating images. Resampling detector algorithms are based on the search for periodic and systematic dependencies between neighboring pixels, as these are inserted when applying the resampling operation. To hide resampling it is necessary to break the periodic equidistance introducing geometric distortions, also known as watermark attacks. In this case, a random distortion vector is overlapped in each pixel position where a parameter determines the distortion degree introduced. To avoid creating visible noise features in the image, the distortion strength should be modulated using two edge detectors, one vertically and the other horizontally.

In [VTT13] the features introduced in the JPEG compression process are studied and a method for detecting JPEG traces is proposed, even when anti-forensics aspects are taken into account in the compression.

In [LWLD13] a method to detect image splicing is proposed. Finally, in [BM13] a complete and detailed survey about passive image forgery detection techniques can be found.

3.3.2 Destruction of Image Identity

In [GKWB07] it was shown that the simple subtraction of wavelet domain characteristics of the images is not sufficient to eliminate noise in an image, and also that this procedure leaves visible traces on the resulting image. There is another well-known method for removing noise from an image called flatfielding, and this method is typically used in astronomy or planes scanning, to improve image quality.

The flatfielding is based on the main components of image noise: the FPN and the PRNU. There are several sources of imperfections and noise introduced at different stages of the creation pipeline of an image. Even if an uniform and fully lighted picture is taken, it is still possible to see small changes in the intensity between pixels. This is due to the shot noise being random and, in large part, the pattern noise being deterministic and being approximately equal if several pictures of the same scene are taken.

The FPN noise is calculated in terms of a dark frame d (Equation 3.2) averaging N images x_{dark} taken in a completely dark environment that can be emulated by completely covering the camera lens. That is to say, the Equation 3.2 computes the dark frame d.

$$d = \frac{1}{N} \sum x_{dark} \tag{3.2}$$

The PRNU noise is calculated in terms of a flatfield f (Equation 3.3) averaging L images $x_{lighted}$ from homogeneously lighted scenes. It is necessary to eliminate the noise FPN from the L images by subtracting the dark frame d before computing the average.

$$f = \frac{1}{L} \sum_{L} (x_{lighted} - d) \tag{3.3}$$

As described in [LFG06] [GKWB07], attackers may try to avoid the correct source identification since it is possible to delete and to remove image fingerprints. The fingerprint subtraction from an image x taken with a specific camera is computed with the Equation 3.4 subtracting a dark current d to the original image x and then dividing the result by the flat frame f. $\tilde{x}$ is the image with its fingerprint subtracted.

$$\tilde{x} = \frac{x - d}{f} \tag{3.4}$$

Despite the fact that the results obtained with this technique are good, it has some drawbacks:

- Performing a perfect flatfielding with a large number of photos is difficult because the parameters to compute the PRNU and FPN must match the parameters from the victim picture.

- The proposal assumes that the attacker can access the source camera of the image x to generate the dark and the flat frames, this scenario is not close to reality.

There are other less robust possibilities to destroy image identity which in some cases may be effective because they do not need extra images from the source camera. However,

instead of this facility the image quality could be reduced and some visual features could be introduced into images. Examples of this kind of technique are: rotating the image a few degrees, scaling the image or applying a Gaussian filter that blurs the image.

In [BK13] PRNU noise is used to pinpoint the camera device which could be undesirable for some users who want to protect their privacy and preserve their anonymity while sharing or spreading images.

In [DSM14] the authors provide an analysis of the seam-carving-based source camera anonymization method by determining the limits of its performance introducing two adversarial models. The results of the analysis shows that the effectiveness of the deanonymization attacks depend on various factors that include the parameters of the seam-carving method, strength of the PRNU noise pattern of the camera, and an adversary's ability to identify uncarved image blocks in a seam-carved image.

In [KDSM15] a technique for circumventing the PRNU based source attribution by mainly focusing on adaptive PRNU denoising method and seam-carving based anonymization is evaluated. Moreover, a panoramic-image-stitching as a means to impede source attribution is introduced.

In [KD15] an improvement on the existing adaptive PRNU denoising method against source camera identification is introduced and anonymization benchmarks with other source anonymization techniques are provided.

3.3.3 Forgery of Image Identity

In the same way that image noise can be removed using the flatfielding technique, it is possible to inject the sensor noise from a different camera using the inverse flatfielding with Equation 3.5 [GKWB07].

$$\tilde{y} = \tilde{x} \cdot f_{forged} + d_{forged} \tag{3.5}$$

where f_{forged} and d_{forged} correspond to the camera that is intended to attack and $\tilde{x}$ is the original image without noise.

In [SLFK10] the Algorithm 1 is proposed to forge the identity of a camera, where $C1$ is the attacker camera, $C2$ is the victim camera and P is a picture taken by $C2$.

Algorithm 1: Forgery of image identity

1. Compute attacker camera $C1$ fingerprint average $F(C1)$;
2. Take a picture P with the victim camera $C2$;
3. Add $F(C1)$ to the picture P;

In case the dimensions of $F(C1)$ and P do not match, a cut or a reconstruction must be applied to match the image sizes. An improvement to the previous falsification algorithm is also proposed by [SLFK10] to mask the features of the camera $C2$. This technique is presented in Algorithm 2.

The subtraction of $F(C2)$ tries to eliminate the correlation between picture P and the camera $C2$, that is to say the existing fingerprint is subtracted before applying the

Algorithm 2: Forgery of image identity with concealed camera

① Compute attacker camera $C1$ fingerprint average $F(C1)$;
② Compute victim camera $C2$ fingerprint average $F(C2)$;
③ Take a picture P with the victim camera $C2$;
④ Subtract $F(C2)$ to P;
⑤ Add $F(C1)$ to the picture P;

attacker camera fingerprint.

[JB14] proposes a technique based on the study of second-order statistics derived from the co-occurrence matrix for detect the presence of counter-forensic attacks.

[HCHY15] proposes an image forgery detection scheme that identifies a tampered foreground or background image using image watermarking and alpha mattes, the proposed method uses (a) component hue difference based spectral matting, (b) image watermarking based on the discrete wavelet transform, discrete cosine transform, and singular value decomposition and (c) the difference between the obtained singular values are used to detect tampering of foreground or background image.

3.4 Summary

In this chapter, the work related to the two main research lines of this thesis has been reviewed: digital image acquisition source identification in both open and closed scenarios, as well as possible attacks that source identification techniques may suffer.

Chapter 4

Image Source Acquisition Identification of Mobile Devices Based on the Use of Features

The identification of the device type or the make and model of image source are two important branches of forensic analysis of digital images. In this chapter, both of these are addressed, with an approach based on feature extraction from image content and the classification using support vector machines.

This chapter is structured into 3 sections. Section 4.1 presents the specification of the proposed technique for image acquisition source identification (source type or the source make and model) based on feature extraction from image content and the different sets of features (Noise, Color, IQM and Wavelets) used by the technique. In Section 4.2, a set of experiments for the identification of device type and the acquisition source identification of the image are performed, and their results are shown. Finally, the chapter ends in Section 4.3 with a brief summary of the above.

4.1 Technique Description

As it has been observed in the chapter of related works there is a wide variety of image features classified into types according to their obtaining base. A feature-based source identification algorithm does not get better results only for the reason of using more features. In fact, it may be the case that using a greater number of features the results tend to worsen. The basis of an algorithm that obtains good results in a conjunction of many factors among which stand out: the choice of features that really determine the image identity, the choice of a suitable number of features and the choice of a good classification method. Unfortunately, often only the experimentation with real images can offer us data about the results of a features based algorithm.

Large numbers of digital images are circulating daily on the Internet or are used as evidence or proof in judicial proceedings. As a consequence, forensic analysis of digital images generated by devices as digital camera, mobile devices, scanner or computer

becomes important in many real-life situations. It is noteworthy that forensics specific images techniques are required for mobile devices, not to be valid in most cases because there are significant intrinsic features which differentiate these types of devices. An example of this is presented in a situation where a forensic analyst needs to identify the type (camera, scanner, computer) or class (make and model) of the image acquisition source.

The technique proposed for image source acquisition identification (source type or source make and model) is based on feature extraction from image content.

There will be a set of images from known sources to be used for training an SVM classifier [HCL03] and another set of images from unknown sources that will be used in the test stage to find out their acquisition source.

The technique can be used to analyze images with different acquisition situations and resolutions, with successful identifications results. In addition, the source identification method proposed is more general because it is useful in a larger set of classification problems.

The main contribution to this technique is a new features generation approach in which the following can be found: sensor pattern noise, color features, image quality metrics and wavelets features. The combination of these features allow the image source identification of images from different types of devices between (images from mobile phones, images obtained from a scanner, and a computer-generated images) and mobile devices of the same brand and different model.

Regarding classification, Michie et al. [MST94] perform a study of different classification methods such as distance-based classifiers, Bayesian classifiers, neural networks, clustering algorithms and SVM classifiers. As can be observed in the review, the use of SVM classifiers is widely used for these purposes. The kernel choice depends, among other factors, on the nature of the data to be classified. This technique will use an SVM classifier with Non-linear RBF kernel, as it is recommended for use when there is no a priori information about the data. The parameters for the SVM are the same as those used in [RCAGSO+13]. Likewise, the option chosen is the most widely used one by the most recent precise works and they present good results. There are many implementations of SVM classifiers; particularly in this work we opted to use the LibSVM library [CL13].

The set of features to be used can be classified into four major groups, depending on the nature of their obtaining:

- Noise features (16 features).
- Color features (12 features).
- IQM (40 features).
- Wavelets (81 features).

A detailed analysis on each of the aforementioned feature sets will be performed below.

4.1.1 Noise Features

The image generation process tends to introduce various defects in them, which will create noise that will be shown in the final image. One type of noise is caused by defects in the CFA matrix, which include hot point defects, dead pixels, pixel traps, column defects and cluster defects. Such defects cause said pixels to differ largely from the rest in the original image; in many cases it makes no difference to have one image or another, since this pixel will always show the same value. For example, dead pixels will appear in the image as black pixels, or hot point pixels will appear as very bright pixels.

The noise pattern in an image refers to any spatial pattern that does not change from one image to another and is caused by a "dark current" and a PRNU [KMC+07]. There are several filters to soften the effect of this noise. The Gaussian filter will be used for several reasons. Its kernel is separable, which allows for fast computation; Gaussian filter removes "high-frequency" components from the image as it is a low-pass filter which helps to reduce noise whilst reducing edge blurring; This filter guarantees a non-negative result (always another valid image); The degree of smoothing is controlled by the parameter σ (larger σ for more intensive smoothing); and also the Gaussian filter produces more uniform smoothing than other filters such as median filter. For these reasons, this filter will be used to eliminate noise in images and then obtain different features.

One of the objectives is to get a set of features that allow us to differentiate between the different types of devices. To do this we firstly take into account that digital cameras use a two-dimensional array sensor whereas most scanners use a linear array sensor. In the case of scanners, the linear arrangement of the sensor moves to generate the entire image, so it is expected to find the periodicity of the sensor noise within the rows of the scanned image. On the other hand, there is no reason to find sensor noise periodicity within the columns of the scanned image. In the case of digital cameras this type of noise periodicity does not exist. This difference can be used as a basis to discriminate between different types of devices. Noise features extraction is based on [KMD09].

Let I an image of $M \times N$ pixels, M as the rows and N as the columns. We denote I_{noise} the noise of the original image and $I_{denoised}$ is the image without noise. Therefore, the noise is obtained using the Equation 4.1.

$$I_{noise} = I - I_{denoised} \tag{4.1}$$

Then, each color component of the image without noise is subtracted to each color component of the original image, with which we obtain noise components of each pixel disaggregated for each color component.

The image original noise I_{noise} can be modeled as the sum of two components, the constant noise $I_{noiseconstant}$ and random noise $I_{noiserandom}$.

For scanners constant noise only depends of the column index, because the same sensor is moved vertically to generate the complete image. The average noise of all columns can be used as a pattern reference $\hat{I}_{noiseconstant}(1, j)$ because the random noise components were cancelled (see Equation 4.2).

$$\hat{I}_{noiseconstant}(1,j) = \frac{\sum_{i=1}^{M} I_{noise}(i,j)}{M}, 1 \leq j \leq N \tag{4.2}$$

As the normalized correlation one of the most commonly used measures in image source identification [Blo08] [Fri09] [Li10a] [CAPI10], for detecting the similarity between different rows with the pattern reference, we use the correlation of these rows with the pattern, as is showed in Equation 4.3.

$$correlation(X,Y) = \frac{X - \overline{X} \ \cdot \ Y - \overline{Y}}{X - \overline{X} \ \cdot \ Y - \overline{Y}} \tag{4.3}$$

Then the same process is performed to detect the similarity of the columns with the pattern reference. After obtaining the correlation between rows and between columns we will go to obtain the feature set. It should be noted at the time of obtaining the features, that in the case of scanners the orientation of the image is critical, because features obtained will be completely different.

For each type of correlation first order statistical values are obtained, which are: mean, median, maximum and minimum. Mode feature was discarded, since after several analysis and experiments was observed it was a useless feature, because when we are dealing with floating values, they did not exist in the majority of the cases repeated values. Tests were performed truncating float values, but the results were not good, decreasing the success rate. Other high order features are variance, kurtosis and skewness. All of them measure statistical values more specifically than previous ones. Also, the ratio features between rows and columns correlations are added. Finally the average noise per pixel feature was included. This feature does not depend on rows or columns correlations with the reference pattern, but is independent and it can distinguish between different types of devices, such as computer generated images.

In total a set of 16 features are obtained: 7 rows features, 7 columns features, the ratio between rows and columns correlations and the average noise per pixel.

4.1.2 Color Features

The configuration of the CFA filters, the demosaicing algorithm and color processing techniques mean that signals in the color bands may contain treatments and specific patterns. In order to determine the differences in color features for different camera models, it is necessary to examine the first and second order statistics of the pictures taken with them. Then, a set of 12 color features based [HLZ10] are proposed.

- **Pixels average value**. For this measure it is assumed that the average values of the RGB channels of an image should be the gray color, as long as the image has enough color variations. This measure is performed for each RGB channels (3 features).

- **Correlation pair between RGB bands**. This measure expresses the fact that depending on the structure of the camera, the correlation between the different

color bands can change. In the implementation of this feature it uses the Pearson correlation coefficient to determine the correlation values between the bands. As a result we obtain three features which come from measuring the correlation between the RG, RB and GB bands.

- **Neighbor distribution center of mass for each color band**: This measure is calculated for each band separately (3 features). Firstly, the total number of pixels for each color value is calculated, obtaining a vector with 256 components. Then, with these calculated values the sum of neighboring values are obtained, that is, for each i value of the vector previously calculated the component $i-1$ and $i+1$ is added. Finally, the center of mass of the latter vector is calculated, which will return a value between 0 and 255.

- **Energy ratios between pairs RGB**. This feature depends on the white dots correction process of the camera. They are 3 features which are defined in Equations 4.4.

$$E_1 = \frac{|G|^2}{|B|^2} \qquad E_2 = \frac{|G|^2}{|R|^2} \qquad E_3 = \frac{|B|^2}{|R|^2} \tag{4.4}$$

4.1.3 Image Quality Metrics

Different camera models produce images of different quality. There may be differences in image brightness, sharpness or quality color. These differences propose a set of quality metrics features that help to distinguish the image source. Image Quality Metrics are of utmost importance in providing quantitative data on the quality of a rendered image [ASS02] [AMS03].

In order to get more detailed difference of images, there are different IQM categories: measures based on the pixels differences, measures based on correlation and measures based on spectral distance.

For obtaining this set of metrics, a filtered image in which the noise of the original image is reduced to perform different calculations is needed in addition to the original image. For the reasons mentioned in Section 4.1.1, a Gaussian filter that allows us to perform image smoothing is used. For obtaining two-dimensional Gaussian kernel we used the Equation 4.5.

$$2\pi\sigma^{2\ -1} * e^{\frac{-\left(i^2+j^2\right)}{2\sigma^2}} \tag{4.5}$$

where i is the distance from the origin in the horizontal axis, j is the distance from the origin in the vertical axis (for example, in the matrix center i and j are equal to 0), and σ is the standard deviation of the Gaussian distribution which represents a threshold or factor value specified by the user.

After the core is obtained, it is normalized, so that the sum of all its components is 1. This is necessary to obtain a smooth image but with the same colors as the original. The normalization is performed dividing each component by the sum of the values of all the components. For obtaining the metrics a filter with a 3×3 kernel with $\sigma = 0.5$ is used. In Equation 4.6 the resulting filter is shown.

$$h = \begin{vmatrix} 0.026625868 & 0.196748265 & 0.026625868 \\ 0.196748265 & 1.453735842 & 0.196748265 \\ 0.026625868 & 0.196748265 & 0.026625868 \end{vmatrix} \tag{4.6}$$

Each pixel of the new image is calculated by performing the neighborhood transformation, applying the Equation 4.7 on original image pixel using the resulting kernel of filter 4.6.

$$\begin{aligned} I'(x,y) &= h(0,0) * I(x-1,y-1) + h(0,1) * I(x,y-1) + h(0,2) * I(x+1,y-1) + \\ & h(1,0) * I(x-1,y) + h(1,1) * I(x,y) + h(1,2) * I(x+1,y) + \\ & h(2,0) * I(x-1,y+1) + h(2,1) * I(x,y+1) + h(2,2) * I(x+1,y+1) \end{aligned} \tag{4.7}$$

It is necessary to consider the edges of the image to make the transformation. In our case we consider an outer edge with pixel value 0.

The measures based on pixel differences calculate the distortion between two images on the basis of their pixel-wise differences. Among these measures are the Minkowski Metrics, Mean Square Error, and Mean Absolute Error (see Equations 4.9 to 4.11, respectively).

The correlation-based measures estimate the similarity between two digital images in terms of the correlation function, and they are complementary to the measures based on pixel differences. In this measure category are Czekonowski Distance, Normalized Cross Correlation, and Structural Content (Equations 4.8, 4.14 and 4.15, respectively).

The spectral distance measures consider the distortion penalty functions obtained from the complex Fourier spectrum of images. The measures grouped in this category are: Spectral Phase, Spectral Magnitude, Weighted Spectral Distance, Median Block Spectral Magnitude, Median Block Spectral Phase, and Median Block Weighted Spectral Distance (Equations 4.20 to 4.22 and 4.26 to 4.28, respectively).

Following the specification of the 40 IQM features based on [HLZ10].

- **Czekonowsky distance**: The Czekonowsky distance is a useful metric for comparing vectors with no negative components as in the case of color images and it is calculated using the Equation 4.8.

$$M = \frac{1}{MN} \sum_{i=1}^{M} \sum_{j=1}^{N} \left(1 - \frac{2 \sum_{k=1}^{3} min\left(C_k(i,j), \hat{C}_k(i,j) \right)}{\sum_{k=1}^{3} \left(C_k(i,j) + \hat{C}_k(i,j) \right)} \right) \tag{4.8}$$

  In this equation and the subsequent $C_k(i,j)$ and $\hat{C}(i,j)$ refer to the pixels in the (m,n) position of the original image and the smoothed image (filtered image with noise reduction) respectively. Furthermore, M and N are the horizontal and vertical size of the image respectively.

- **Minkowsky metrics**: Minkowsky metrics for $\gamma = 1$ and $\gamma = 2$ are based on Equation 4.9.

$$M_\gamma = \frac{1}{K} \sum_{k=1}^{K} \left\{ \frac{1}{N^2} \sum_{i,j=1}^{N} C_k(i,j) - \hat{C}_k(i,j) \right\}^{\gamma} \right\}^{\frac{1}{\gamma}} \tag{4.9}$$

This equation calculates the norm L_γ of dissimilarity between two images, where N^2 is the total number of pixels. In this formula and ongoing, k will refer to each of the image channels. It must be taken into account that this formula performs the average of Minkowski metric for all channels of the image.

$\gamma = 1$ is corresponding with the *Mean Absolute Error* (MAE) and $\gamma = 2$ with the *Mean Square Error* (MSE) (estimated by Equations 4.10 y 4.11, respectively). In both cases, high values of MAE or MSE correspond with low quality images.

  - *Mean Absolute Error*:

$$MAE = \frac{1}{MN} \sum_{m=1}^{M} \sum_{n=1}^{N} C_k(i,j) - \hat{C}_k(i,j) \tag{4.10}$$

  - *Mean Square Error*:

$$MSE = \frac{1}{MN} \sum_{m=1}^{M} \sum_{n=1}^{N} C_k(i,j) - \hat{C}_k(i,j) \Big)^2 \tag{4.11}$$

These metrics are applied to each of the bands separately, so that three features for the MAE and others three for the MSE are obtained

- ***Laplacian Mean Square Error* (LMSE)**: This metric is based on the importance of measuring the edges and is defined by Equation 4.12. A LMSE high value indicates that image quality is poor. It is defined as follows:

$$LMSE = \frac{\sum_{m=1}^{M} \sum_{n=1}^{N} \Big[L(x(m,n)) - L \quad x^{(m,n)} \Big)\Big]^2}{\sum_{m=1}^{M} \sum_{n=1}^{N} [L(x(m,n))]^2} \tag{4.12}$$

where $L(x(m,n))$ is the Laplacian operator estimated by Equation 4.13.

$$\begin{aligned} L(x(m,n)) &= x(m+1,n) + x(m-1,n) + \\ &\quad x(m,n+1) + x(m,n-1) - 4x(m,n) \end{aligned} \tag{4.13}$$

- **Normalized Cross Correlation**: The closeness between two digital images can also be quantified in terms of a correlation function. The quality metric of the

normalized cross-correlation measurement for each image band k is defined as the Equation 4.14:

$$NCC = \frac{\sum_{i,j=0}^{N-1} C_k(i,j) * \hat{C}_k(i,j)}{\sum_{i,j=0}^{N-1} C_k(i,j)^2} \tag{4.14}$$

- **Structural Content**: The structural content of an image quality metric is defined for each band k with the Equation 4.15.

$$SC = \frac{\sum_{i,j=0}^{N-1} C_k(i,j)^2}{\sum_{i,j=0}^{N-1} \hat{C}_k(i,j)^2} \tag{4.15}$$

- **Spectral Measures**: To determine these measures the *Discrete Fourier Transform* (DFT) of the original image and the smoothed image, denoted as $\tau_k(u,v)$ and $\hat{\tau_k}(u,v)$ for a band k, are defined by Equations 4.16 and 4.17, respectively.

$$\tau_k(u,v) = \sum_{m,n=0}^{N-1} C_k(m,n) * e^{\left[-2\pi i m \frac{u}{N}\right]} * e^{\left[-2\pi i n \frac{v}{N}\right]} \tag{4.16}$$

$$\hat{\tau}_k(u,v) = \sum_{m,n=0}^{N-1} \hat{C}_k(m,n) * e^{\left[-2\pi i m \frac{u}{N}\right]} * e^{\left[-2\pi i n \frac{v}{N}\right]} \tag{4.17}$$

where $u = 0, \ldots, M-1$, $v = 0, \ldots, N-1$ and (u,v) are the coordinates of an image pixel in transform domain. Equations 4.18 and 4.19 respectively define the phase and magnitude of the DFT spectrum.

$$\varphi(u,v) = \arctan(\tau_k(u,v)) \tag{4.18}$$

$$M(u,v) = |\tau_k(u,v)| \tag{4.19}$$

With the above concepts the following image quality metrics can be defined for each image band: Spectral Phase, Spectral Magnitude and Weighted Spectral Distance (Equations 4.20 to 4.22).

– *Spectral Phase*:

$$SM = \frac{1}{MN} \sum_{u=1}^{M} \sum_{v=1}^{N} |\varphi(u,v) - \hat{\varphi}(u,v)|^2 \tag{4.20}$$

– *Spectral Magnitude*:

$$SP = \frac{1}{MN} \sum_{u=1}^{M} \sum_{v=1}^{N} M(u,v) - \hat{M}(u,v)^2 \tag{4.21}$$

– *Weighted Spectral Distance*: Performs a weighted average of the phase and magnitude spectrum.

$$WSD = \rho * SM + (1-\rho) * SP \tag{4.22}$$

where for this case $\rho = 2.5 * 10^{-5}$.

These characteristics can also be obtained for each image block. For this we consider that the image is divided in L blocks with bxb size, and then the above features are calculated. In this way the following l-th block features for each band of the block can be defined by Equations 4.23 to 4.25:

$$J^l{}_{\varphi} = \left(\sum_{u,v=0}^{b-1} \varphi^l(u,v) - \hat{\varphi}^l(u,v)^{\gamma} \right)^{\frac{1}{\gamma}} \tag{4.23}$$

$$J^l{}_{M} = \left(\sum_{u,v=0}^{b-1} M^l(u,v) - \hat{M}^l(u,v)^{\gamma} \right)^{\frac{1}{\gamma}} \tag{4.24}$$

$$J^l = \rho * J^l{}_{M} + (1-\rho) * J^l{}_{\varphi} \tag{4.25}$$

For calculating these features we have used $\gamma = 2$ and a 32×32 block size. After calculating these measures for each block we can get the following features: Median Block Spectral Magnitude, Median Block Spectral Phase, Median Block Weighted Spectral Distance (Equations 4.26 to 4.28).

– *Median Block Spectral Magnitude*:

$$\text{MBSM} = \text{median}\, J^l{}_{M},\ \text{l} = 1, ..., \text{L} \tag{4.26}$$

– *Median Block Spectral Phase*:

$$\text{MBSP} = \text{median}\, J^l{}_{\varphi},\ \text{l} = 1, ..., \text{L} \tag{4.27}$$

– *Median Block Weighted Spectral Distance*:

$$\text{MBWSM} = \text{median}\, J^l,\ \text{l} = 1, ..., \text{L} \tag{4.28}$$

- **Measures based on the human visual system**: Images can be processed by filters which simulate the *Human Visual System* (HVS). One of the models used for

this is a band-pass filter with a transference function in polar coordinates defined by Equation 4.29.

$$H(\rho) = \left\{ \begin{array}{cc} 0.05e^{\rho^{0.554}} & \rho < 7 \\ e^{-9[|log_{10}\rho - log_{10}9|]^{2.3}} & \rho \geq 7 \end{array} \right\} \tag{4.29}$$

where $\rho = \sqrt{(u^2 + v^2)}$. Equation 4.30 defines the operator U.

$$U\left\{C\left(i,j\right)\right\} = DCT^{-1}\left\{H \quad \sqrt{u^2+v^2} \quad \omega\left(u,v\right)\right\} \tag{4.30}$$

where $\omega\left(u,v\right)$ denotes the two-dimensional DCT of the image and DCT^{-1} is the inverse two-dimensional DCT.

Finally the image quality metrics that we obtain for each band of the image based on these measures are Normalized Absolute Error and HVS based L2 (Equations 4.31 and 4.32, respectively).

– *Normalized absolute error*:

$$NAE = \frac{\sum_{i,j=0}^{N-1} U\left\{C_k\left(i,j\right)\right\} - U\left\{\hat{C}_k\left(i,j\right)\right\}}{\sum_{i,j=0}^{N-1} \left|U\left\{C_k\left(i,j\right)\right\}\right|} \tag{4.31}$$

– *HVS based L2*:

$$L2 = \left\{ \frac{1}{N^2} \sum_{i,j=0}^{N-1} U\left\{C_k\left(i,j\right)\right\} - U\left\{\hat{C}_k\left(i,j\right)\right\}^{\,2} \right\}^{\frac{1}{2}} \tag{4.32}$$

The HVS is too complex, however, the incorporation of a simplified HVS model into objective measures has been reported to lead to better correlations [Wat93] [NB92] [FBA97] [Nil85]. So the measures based on HVS gain importance.

4.1.4 Wavelet Features

Due to the deterministic property of the sensor pattern noise which is present in an image, this pattern can be used as a footprint to identify the device that generated the image under investigation. It can be said that the sensor pattern noise is to a digital camera as a fingerprint is to a human being.

In contrast to the Fourier Transform which represents the signals as a sum of sinusoidal waves that are not localized in time and space, the Wavelet Transform is more convenient in the analysis of signals with abrupt changes as images since their functions are located in time and space. There are a large number of alternatives of wavelet functions (Haar, Daubechies, Coiflet, Symlet, Meyer, etc.) for analyzing the signals, allowing the choice

of the base of functions whose form is better approximated to the characteristics of the signal to be analyzed. Based on the results of previous work [RCAGSO+13] [SOGVAG+15] Daubechies functions are the ones that have obtained better results in the extraction of sensor noise.

The Wavelet Transform allows the information to be separated in the manner of frequencies. In this way, it is possible to analyze and / or modify only the information of the frequencies that are of some particular interest, such as sensor noise in order to extract its characteristics.

To identify the acquisition source we require an algorithm that allows us to extract the sensor noise and another that allows us to obtain the features of the fingerprints obtained in order to classify and identify them.

To extract sensor noise the algorithm presented in [AG15] is used.

Finally, a total of 81 features (3 channels × 3 wavelet components × 9 central moments) are calculated using algorithm 3.

Algorithm 3: Extracting features

Input: Sensor fingerprint I_{noise}
Result: 81 features

1 **procedure** ExtractFeatures(I)
2 Separate R, G and B color channels of I_{noise};
3 **foreach** *color channel* **do**
4 Apply a wavelet decomposition in 1 level;
5 **foreach** *component* $c \in \{H, V, D\}$ **do**
6 Compute k central moments with $m_k = \frac{1}{n} \sum_{i=1}^{n} |c_i - \overline{c}|^k$;
7 **end procedure**

4.2 Experiments and Results

This section shows the results of the experiments conducted to identify the type of source device and for source acquisition identification.

It should be noted that the classification of images to be performed in this work be done on what can be called a closed set of elements, i.e., the classes of the elements used in training are the same classes as those used in the test. The images used in the training stage are not used in the testing stage.

4.2.1 Source Device Type Identification

In this experiment we will use an image set composed of: images from mobile phones, images obtained from a scanner, and a computer-generated images. 200 images are used from each set, 100 for the SVM training and 100 for testing. All images have a resolution higher than 1024×768. There is no restriction on the content of the image or the camera configuration parameters at the time of the acquisition.

Images from mobile phones have been obtained from known phones, so the origin of the source can be ensured. Images from 12 smartphones, some of them from the same manufacturer, were selected.

For images from scanners and computer-generated images, our own sources and the Flickr website were used. The set of images downloaded from the web had a set of filters applied to them in order to obtain a set with higher reliability that would introduce the least possible noise into the experiments. All images downloaded from *Flickr* are originals with no resizing. As a second filter for scanned images, those which had the tag "scanned images" and made reference to a retail scanner model were used. For computer-generated images, we discarded the images that had a "camera model" tag with a value from a retail scanner or camera. 15 scanned images were selected, some of which were chosen from the same manufacturer. With respect to computer-generated images, precise information on the number of applications or type of computers used cannot be indicated.

As can be seen, there is a high number of different kinds of devices (makes and models) of the three types, which greatly hampers the classification.

For the experiments we have taken into account the following configuration parameters: size of crop applied to the image, crop position (centered or upper-left corner) and application of different feature sets (Noise Features, Color Features, IQM Features and Wavelet Features).

Table 4.1 shows the results of success rates and the configuration parameters used in the 10 experiments.

Table 4.1: Source device identification between mobile camera, scanner and computer generated images

Features	**Crop Size**	**Crop Align**	**Device**			**Average**
			Camera	**Computer**	**Scanner**	
Noise	Full Size	-	70	54	57	**59.95%**
Noise	1024×768	Center	66	80	46	**62.39%**
Noise	800×600	Center	76	60	49	**60.68%**
Noise	640×480	Center	62	61	48	**56.62%**
Noise	1024×768	Upper-left corner	76	59	40	**56.40%**
Noise	800×600	Upper-left corner	65	38	44	**47.72%**
Noise	640×480	Upper-left corner	74	54	37	**52.88%**
All Features	1024×768	Center	66	73	72	**70.26%**
All Features	800×600	Center	69	74	71	**71.30%**
All Features	640×480	Center	77	73	63	**70.75%**
Average			**69.91%**	**61.35%**	**51.42%**	**60.42%**

From the analysis of the results, general and specific conclusions about the various configurations used in each experiment can be obtained. Encompassing all the experiments, it is observed that success rates are not excessively high (60.42% on average and 71.30% in the best case); it can be concluded that this technique is not particularly suitable for this purpose. It is important to emphasize, as noted above, that the number

of different makes and models used for this experiment is high, which predictably causes success rates to drop. That being said, it should be noted that this study does provide interesting results on the configuration parameters used, since between the best and the worst result there is a difference in the average success rate of 23.48%.

In general it can be concluded that the only use of the noise features do not perform well for identifying the source type when the number of devices to be classified is high, since the average success rate of all experiments is 56.65%. Since the results are not good then a set of experiments reducing the number and types of devices will be made to observe their results.

The results improve significantly when all the features to identify the source type are used. Given the high number of classes, the results can be qualified as acceptable, since the average success rate for all experiments carried out using these features is 70.77%. Also, as can be observed in the results, we conclude that the crop size affects the results: the smaller the crop, the lower the success rate, even if the differences are not extremely significant. It is also noteworthy that with a 1024×768 crop size better results are obtained than when using the full-sized image, i.e., from a given crop size, the results get worse.

A series of experiments will be performed reducing the number of types to be classified, in order to test the behavior of the results when we only use noise features, since these are the ones that obtained the worst results previously. The results are shown in Table 4.2.

Table 4.2: Source type identification with noise features

Device Type 1	Device Type 2	Features	Crop Size	Crop Align	Average
Scanner	Smartphone	Noise	1024×768	Center	95.79
Scanner	Smartphone	Noise	640×480	Center	96.16
Scanner	Smartphone	Noise	400×300	Center	96.73
Computer	Smartphone	Noise	1024×768	Center	79.96
Computer	Smartphone	Noise	640×480	Center	79.76
Computer	Smartphone	Noise	400×300	Center	78.55
Computer	Scanner	Noise	1024×768	Center	82.87
Computer	Scanner	Noise	640×480	Center	81.10
Computer	Scanner	Noise	400×300	Center	80.91

As can be seen, the success rate goes up considerably as it was expected, reaching 85.44% on average. When the number of types of devices is reduced to two and as a result the number of classes is reduced the results are acceptable.

The first general conclusion obtained corroborates an earlier conclusion, since it is observed that the crop size does not significantly affect the results. The best results are those that distinguish between smartphone and scanned images, with 96.23% average success rate. The second best result appears with the distinction between scanned and computer-generated images, with 81.62% average success rate. The worst result was obtained in the distinction between computer-generated and smartphone images, with 79.42% average success rate. Still, any of the results of these experiments are significantly

better than the results in Table 4.1. Therefore, it can be concluded that in general, the use of noise features for type of source distinction only obtains acceptable results when the number of classes is not high.

4.2.2 Image Source Identification for Mobile Devices

Given the importance of mobile images today, below we will show the experiments performed to identify the acquisition source of images from mobile devices, i.e., the classification of an image set according to the make and model of the camera that generated them.

In these experiments a set of 200 images will be used, 100 for the SVM training and 100 for testing. 12 smartphone models were used: IPhone 4s (I1), IPhone 5s (I2), Blackberry 8520 (BB), Huawei U8815 (HU), LG E400 (LG1), LG P760 (LG2), Nokia 800 (N1), Samsung GT-I9001 (S1), Samsung GT-I9100 (S2), Samsung GT-I8160P (S3), Samsung GT-5830M (S4) and Sony C2105 (SE1). The images comply with the same restrictions as the cameras in the previous section. The mobile device digital cameras used and their configurations are showed in Table 4.3.

Table 4.3: Configurations used in mobile device digital cameras

Brand	Model	Resolution	Taking Conditions
Apple	IPhone 4s (I1)	2 MP (2048×1536)	Scene type: Any Orientation: Vertical Flash: Disabled Light: Natural White balance: Auto Digital zoom ratio: 0 Exposure time: 0 seg ISO speed: Auto
	IPhone 5s (I2)	3.15 MP (2048×1536)	
Black Berry	8520 (BB)	2 MP (1600×1200)	
Huawei	U8815 (HU)	2 MP (1600×1200)	
LG	E400 (LG1)	2 MP (1600×1200)	
	P760 (LG2)	3.15 MP (2048×1536)	
Nokia	800 (N1)	2 MP (1600×1200)	
Samsung	GT-I9001 (S1)	2 MP (1600×1200)	
	GT-I9100 (S2)	2 MP (1600×1200)	
	GT-I8160P (S3)	5 MP (2592×1944)	
	GT-5830M (S4)	5 MP (2592×1944)	
Sony Ericsson	C2105 (SE1)	2 MP (1600×1200)	

The experiments have been grouped into 3 groups with the aim of obtaining conclusions on: the use of different feature sets, crop size, the number of devices used for the classification, and the use of devices from the same manufacturer. The experiments where all devices are from the same manufacturer put the techniques presented to the test. Hardware and software components of the cameras from the same manufacturer are generally very similar or even the same, which obviously presents serious difficulties or impossibility of distinction among the different smartphone models.

Table 4.4 shows the first set of experiments in which 7 models of mobile devices from different manufacturers are used. Different types of combinations of features sets were tested. Most experiments were performed with a crop size of 1024×768, since as this is considered a large enough size to obtain good results, as shown in the previous experiments.

Table 4.4: Image source acquisition identification for 7 smartphones

Features	Crop Size	Crop Align	I1	HU	LG2	N1	BB	S1	SE1	Average
All Features (Daubechies 8-tap)	1024×768	Center	93	96	80	94	91	70	85	86.54%
Noise	1024×768	Center	41	42	35	18	40	40	62	37.67%
Color	1024×768	Center	24	37	20	40	31	19	44	29.27%
IQM	1024×768	Center	13	88	46	89	7	34	2	21.65%
Wavelet Daubechies 8-tap	1024×768	Center	95	96	96	94	92	76	93	91.46%
Wavelet Haar	1024×768	Center	95	87	97	70	86	56	91	81.84%
Color + IQM + Wavelet Daubechies 8-tap	1024×768	Center	93	94	90	90	90	53	85	83.67%
All Features (Daubechies 8-tap)	800×600	Center	91	96	84	92	95	56	85	84.41%
All Features (Daubechies 8-tap)	640×480	Center	90	95	84	89	88	51	88	82.15%

The experiment reveals that noise, color and IQM feature sets are individually completely invalid, since the best result obtains an 37.67% average success rate, which is unacceptable. With the remaining set of features (wavelets), two experiments were conducted using different types of wavelet: Daubechies 8-tap and Haar. The results show that Daubechies 8-tap obtains better results than Haar and the best results of all experiments (91.46%).

With respect to the different feature combinations, it is observed that when we use all the features good results are obtained (86.54% in the best case), since, although they are slightly worse than the best result, the difference is not very significant (4.92%). Also, the success rate when all the features are used subtly drops the smaller the crop size gets.

The combination of all the features except noise features, which are mainly focused on identifying the source type, yields an average success rate of 83.67%. These results, even if not bad, are far from those obtained with the wavelets and worse than when the combination of all features is used.

In order to further evaluate the results of using all feature sets, in the next set of experiments 10 models of mobile devices, some of them from the same manufacturer are used. The results of the experiments are shown in Table 4.5.

As previously stated, the fact that there are devices from the same manufacturer and similar features greatly hampers the classification task, since cameras can be identical or virtually identical. As expected, we find that the larger the number of devices, some of them having the same manufacturer, lowered success rates in all cases (6.56% in the best case). However, it is considered that the decrease is not extremely pronounced, considering that there are 3 more devices and particularly 3 pairs of devices from the same manufacturer. It is important to note that the LG E400 device has in all cases the lowest success rates by far compared to the other devices (43.71% average success rate). 31%, 38%, 38% and 32% of the images from the LG E400 (Optimus L3) were classified as images from the LG P760 (Optimus L9) in the first, second, third and fourth experiments

in this set respectively. This clearly indicates a great level of confusion between the LG E400 and LG P760 images, which significantly lowers the overall success rate of each experiment. From the results it can be deduced that the technology and hardware and software components of both cameras could be similar (even if there are two intermediate models, the Optimus L5 and L7), or the defined feature set does not allow us to properly discern between the two cameras. It is also found, as in the previous experiment, that the Haar wavelet is not suitable for classifying images from mobile devices.

Table 4.5: Image source acquisition identification for 10 smartphones

Features	**Crop Size**	**Crop Align**	**I1**	**I2**	**HU**	**LG1**	**LG2**	**N1**	**BB**	**S1**	**S4**	**SE1**	**Avg (%)**
All Features (Daubechies 8-tap)	1024×768	Center	91	87	96	47	89	92	95	61	79	80	79.98
All Features (Haar)	1024×768	Center	84	76	81	43	78	64	64	41	82	89	68.04
All Features (Daubechies 8-tap)	800×600	Center	91	85	95	43	80	88	97	52	72	82	76.23
All Features (Daubechies 8-tap)	640×480	Center	92	77	99	42	84	90	90	45	73	84	74.86

For investigating deeper on the results of distinction between cameras from the same manufacturer the next set of experiments was made, in which we use 4 mobile device models of Samsung manufacturer. The results of the experiments are shown in Table 4.6.

Firstly we observe that the results using all features except noise are worse than those obtained when using all the features, as we could see in the experiments of Table 4.4. Having said this, it is observed that the results are quite good, because in the best case we obtain a 80.69% average success rate. The results obtained for the Samsung GT-I8160P are worse compared with the results of the other devices, in particular a 59.90% average success rate, when the average success rate in the worst case for the rest of the devices is 81.71%. The 20%, 33%, 24% and 23% of the images of the Samsung GT-I8160P (Galaxy Ace 2) were classified as images of the Samsung GT-S5830M (Galaxy Ace) for the first, second, third and fourth experiment of this set, respectively. In fact, this concrete result, reduce a 5.16% the best success rate obtained. Similarly to the previous case, can be supposed the same conclusions, although in this case the similarity of the cameras in all levels has more sense, since there are not intermediate models between the two devices in Ace Samsung series, that is, one succeeds the other.

Table 4.6: Image source acquisition identification for 4 mobile devices of the same manufacturer

Features	Crop Size	Crop Align	S1	S2	S3	S4	Average
All Features (Daubechies 8-tap)	1024×768	Center	93	84	67	81	80.69%
IQM + Color + Wavelet Daubechies 8-tap	1024×768	Center	88	84	50	84	74.65%
All Features (Daubechies 8-tap)	800×600	Center	89	81	61	86	78.42%
All Features (Daubechies 8-tap)	640×480	Center	88	78	63	77	75.96%

The information from these experiments can be a starting point for further works to optimize the success rate for the different cases presented with mobile device images. The choice of feature sets that offer better results and the remaining configuration parameters of the technique must be taken into account.

4.3 Summary

The main objective of this chapter has been to present a technique with the aim of identifying the device type (scanner, computer, mobile camera) or class (make and model) of the image acquisition source, the last specifically for mobile devices. First, the features of Noise, Color, IQM and Wavelets used by the proposal were shown. After, a total of 36 experiments classified into 5 sets, in order to test different configurations of the techniques were described. In the configuration of the experiments, the future use of the technique by the forensic analyst in real situations to create experiments with high technical requirements was taken into account, amongst other things. Finally, we have shown the main results of the identification technique proposed.

Chapter 5

Smartphone Image Clustering

In this chapter a clustering algorithm is proposed. As elements for classification we use a set of features obtained from SPN noise. Broadly speaking, the main difference compared to other techniques is that this proposal takes into account the evolutionary process of cluster formation when calculating the coefficient that determines the cohesion between the elements of the same cluster and separation between different clusters that are being generated. This chapter is divided into four sections. Section 5.1 briefly presents an overview of the types of scenarios in which the techniques for identifying the acquisition source are applied. The proposed algorithm is then specified in detail in Section 5.2. The experiments with image banks and their results are presented in Section 5.3. Finally, the chapter ends in Section 5.4 with a brief summary of what is presented in it.

5.1 Overview

There are two major approaches regarding source acquisition identification: closed scenarios and open scenarios.

A closed scenario is one in which the image source identification is performed on a specific and known beforehand set of cameras. For this approach a set of images from each camera is normally used to train a classifier and later the image source acquisition under investigation is predicted. The most commonly used technique for the digital imaging classification task is SVM, although there are other options, such as the use of neural networks.

This algorithm focuses on image source acquisition identification in open scenarios, i.e., the forensic analyst does not know a priori the camera set to which images whose source identification will be identified belong. Obviously, in this type of classification in which data from cameras are not known beforehand, the objective is not to identify the make and model of the images, but to be able to group the different images into disjoint sets in which all their images belong to the same device.

This approach is very close to real-life situations, since in many cases the set of cameras to which a set of images may belong is completely unknown to the analyst. In addition, it is virtually impossible to have a set of images to train a classifier with all mobile device cameras existing in the world. In this case, being able to group images into sets that

belong to the same device is very useful, as this can provide very valuable and in some cases conclusive information to judicial investigators.

5.2 Technique Description

The proposed unsupervised clustering algorithm is based on the one proposed in [CAPI10]. It is a combination of a hierarchical clustering, and a flat clustering. That is, despite forming a dendogram structure with each iteration of the algorithm, at the end the clusters are taken as unrelated entities since each of them must correspond to a specific device. The general structure of the proposed clustering algorithm is shown in Figure 5.1 (N is de number of images and q is the number of iteration and it begins in 0).

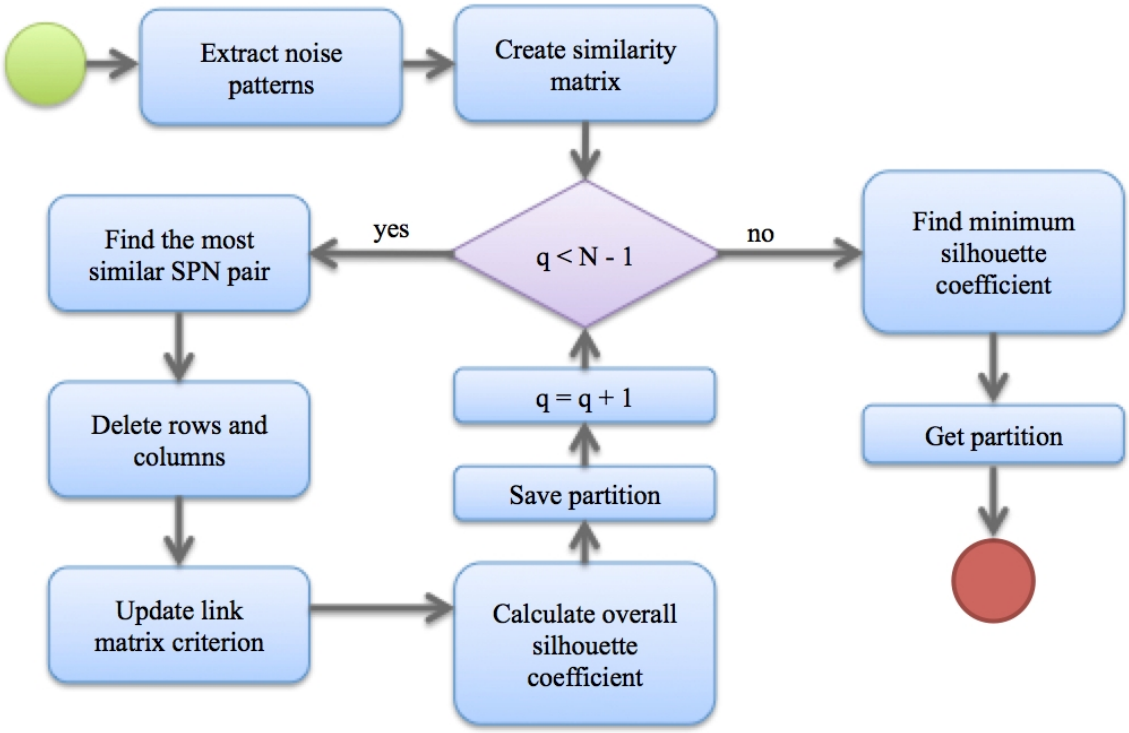


Figure 5.1: Clustering algorithm structure

Prior to performing the clustering, it is necessary to obtain SPNs of the image set I using the extraction algorithm and the parameter of noise suppression $s_0 = 5$ proposed in [AG15]:

$$n^{(i)} = I^{(i)} - F \quad I^{(i)} \tag{5.1}$$

where $i = 1, ..., N$, N is the number of images, $n^{(i)}$ is the noise pattern of each image i, $I^{(i)}$ is the image with sensor noise of each image i, F is the noise removal filter based on wavelet transform. For this, the algorithm developed in [GFF09] was used. No noise improvement algorithm, such as those proposed by [CAPI10] and [Li10b], has been used in our proposal. The Wiener filter in the frequency domain is sufficient to remove most of

the scene details that are present when extracting the SPN.

For each of the N noises $(n_1, ..., n_N)$ the correlation value is obtained using Equation 3.1 and this generates a similarity matrix H of $N \times N$. This matrix is symmetric and consists of ones in its main diagonal (since the correlation of noise with itself is 1). Once the matrix has been generated it will not be necessary to recalculate the correlations between noises along the clustering algorithm, saving time and processing power.

The selected hierarchical clustering algorithm involves finding within the H matrix the noise pair k and l with a highest correlation value. It is worth mentioning that the correlation values in the main diagonal are not taken into account. Then the rows and columns k and l are deleted and both a new row and a new column are added to the matrix. These new row and column values are the result of a linkage criterion. The function chosen for this work was the average linkage method since its results are more satisfactory than with other linkage methods such as single linkage or complete linkage, as is suggested in [CAPI10]. Equation 5.2 shows the function of the average linkage method between two clusters A and B.

$$H(A, B) = \frac{1}{\|A\| \, \|B\|} \sum_{n_i \in A, n_j \in B} corr(n_i, n_j) \tag{5.2}$$

where the $corr(n_i, n_j)$ value is calculated with Equation 3.1 and can be taken from the matrix H to simplify the computational processing. $\|A\|$ and $\|B\|$ is the cardinality of the A and B clusters respectively.

Each iteration of the algorithm takes the two clusters with the highest correlation value in the matrix and mixes the objects contained in them to create a new cluster, while storing the state of the different clusters in partition $P_0, ..., P_{N-1}$ with the aim of knowing the contents of the cluster at any time. In the hierarchical clustering, the final result of the algorithm is a cluster containing all objects. However, in this work each cluster should represent a device at the end of the execution. For this reason, the silhouette coefficient as a measure of validation of clusters was used. The silhouette coefficient measures the similarity index between the elements of a single cluster (cohesion) and the similarity between the elements of a cluster with respect to the others (separation). Unlike Caldelli *et al.* [CAPI10], in our proposal the calculation of the silhouette coefficient is performed for each cluster contained in the P_i partition and not for each pattern noise, as noted in Equation 5.3.

$$s_j = \max(b_j) - a_j \tag{5.3}$$

where a_j (cohesion) is the average correlation between all noise patterns within the c_j cluster and b_j (separation) is the average correlation of noise patterns contained in the c_j cluster with respect to noise patterns in the remaining clusters. The nearest neighboring cluster is taken, namely the one with the highest correlation.

As can be seen the method of calculating the silhouette coefficient varies significantly regarding to the proposal of [CAPI10]. In our proposal for each algorithm iteration as many silhouettes coefficients as exist formed clusters in that iteration are calculated and not as many silhouette coefficients as total images there are. According to the algorithm

progresses and the clusters are increasing their number of images, in our proposal fewer silhouettes coefficients are calculated. Furthermore and the most important thing is that while clusters are formed in each algorithm iteration, each cluster is taken as a single entity for the calculation of silhouette coefficient. Therefore do not take into account each of the independent noises that form each cluster, since what it wants to measure is the cohesion and separation between clusters and not between independent images. Once it has the idea of a cluster as a unitary entity to calculate the silhouette coefficient, it is calculated for each cluster taking into account the obtaining of the maximum separation from other clusters and a high cohesion of all the elements of the formed cluster, as it can see in the Equation 5.3.

For each iteration q of the algorithm a global measure of all the silhouette coefficients calculated from the K clusters is obtained, this is equivalent to averaging the s_j values in q. Equation 5.4 shows this calculation.

$$SC_q = \frac{1}{K} \sum_{j=1}^{K} s_j \tag{5.4}$$

Upon completion of the hierarchical clustering, the SC_q with the lowest value is searched for, which indicates that the partition P_q^* clusters are at a greater correlation level. The number of clusters at that moment should correspond to the actual number of devices. The aim of storing the partition at each time of the algorithm is to avoid rerunning the clustering because information of all the clusters in each iteration q is known. Algorithm 4 shows the proposal's pseudocode.

Algorithm 4: Clustering algorithm

```
1  Calculate n^(i) of each image where i ∈ 1, ..., N;
2  Generate the similarity matrix H ∈ R^(N×N);
3  foreach q ∈ 1, ..., N − 1 do
4      Find cluster H(k, l) with the highest similarity;
5      Remove the pair of rows and columns corresponding to clusters k and l;
6      Calculate the values of the new cluster using average link criteria and add
7      the row and its corresponding column;
8      Determine the overall silhouette coefficient SC_q;
9      Store the partition P_q;
10 Find the partition where min_q(SC_q);
```

As mentioned above, the goal of clustering is to group objects in an unsupervised environment (closed scenario); however, in the chosen methodology it is possible to carry out a training stage and a classification stage to reduce computational complexity and therefore to reduce the execution time of the algorithm. For this it is necessary to divide the image set into two subsets: one of I_e training and one of I_c, classification, the training subset is processed by the algorithm previously described to get to the final K clusters that represent each device. A c_j centroid is then calculated for each cluster by averaging all m noise patterns contained in it. Next the correlation value between each SPN from

the I_c subset and each c_j, centroid is obtained, the image is then classified into the cluster where there has been the highest correlation value. The proposed classification can be observed in Algorithm 5.

Algorithm 5: Clustering algorithm with training stage

1. Calculate the centroid c_j of each cluster where $j \in 1, ..., K$ and $c_j = \frac{1}{m} \sum_1^m n$;
2. Calculate the pattern noise n_i of the classification subset $I_c \subset I$;
3. **foreach** $n_i \in i_c$ **do**
4. Classify n_i in the cluster with the highest correlation with $f_j = \arg\max_j corr(n_i, c_j)$;

5.3 Experiments and Results

The experiments were performed with a total set of 1350 photographs from 9 different mobile device camera models. The total set contains 150 photographs from each model. 6 devices are from different manufacturers (Apple iPhone 5, Huawei U8815, LG E400, Samsung GTS5830M, Zopo ZP980 and Nokia 800 Lumia) and the 3 remaining devices were manufactured by Sony (Sony ST25a, Sony ST25i and Sony C2105).

All the images were cropped to 1024×1024 pixels because the images have different dimensions and working with these at full size it would be computationally more complex. To reduce the degree of error in the grouping all images have a horizontal orientation; it was necessary a 90° rotation images captured in vertical position. The scenes of the photographs were chosen randomly, both indoors and outdoors, and they were also taken at different times and places in order to simulate a more realistic scenario. In the extraction of the noise pattern from all images, the zero-mean of rows and columns was used, 3 RGB color channels were converted to a single matrix in grayscale. Additionally, all experiments were conducted using the Wiener filter in the frequency domain. In Figure 5.2 a diagram of the preprocessing performed on the images is shown.

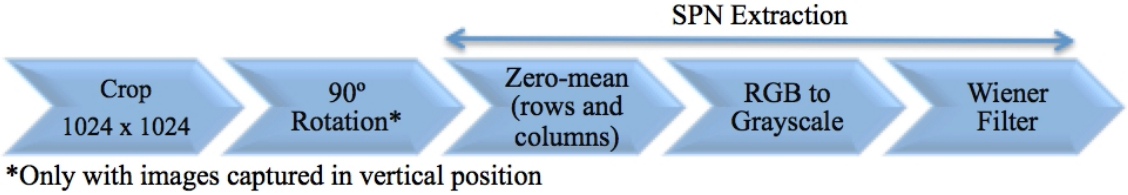


Figure 5.2: Image preprocessing scheme

To measure the degree of certainty in the results, the true positive rate TPR was used. The mean TPR for each of the following experiments is calculated, computing for each cluster the number of photos that have been well classified (TPR of each cluster) and averaging the TPRs of all the resulting clusters (if there are fewer clusters than devices the average takes into account the number of devices). To calculate the TPR of each cluster,

the device that has the largest number of images with respect to the total of images by device needs to be identified within the cluster, that being the predominant device cluster, then calculate the percentage of photos that have been well classified for that device in the cluster. Actually, in the vast majority of cases it can be seen that a cluster is associated with one or more devices, as it can observed in matrices such as the ones in Tables 5.1, 5.2 and 5.3.

Table 5.1: TPR with equal number of devices than cluster

Brand - Model	**Cluster 1**	**Cluster 2**	**Cluster 3**	**Cluster 4**	**Cluster 5**	**Average TPR**
Apple - Iphone 5	49	0	0	1	0	
Huawei - U8815	0	50	0	0	0	
LG - E400	0	1	49	0	0	
Nokia - 800 Lumia	0	0	0	50	0	
Samsung - GT5830m	0	0	0	0	50	
TPR by cluster	**98%**	**100%**	**98%**	**100%**	**100%**	**99.2%**

Table 5.2: TPR with less number of devices than clusters

Brand - Model	**Cluster 1**	**Cluster 2**	**Cluster 3**	**Cluster 4**	**Average TPR**
Apple - Iphone 5	100	0	0	0	
Huawei - U8815	0	100	0	0	
LG - E400	0	0	97	3	
TPR by cluster	**100%**	**100%**	**97%**	**0%**	**99%**

Table 5.3: TPR with more number of devices than clusters

Brand - Model	**Cluster 1**	**Cluster 2**	**Cluster 3**	**Cluster 4**	**Average TPR**
Apple - Iphone 5	100	0	0	0	
Huawei - U8815	0	100	0	0	
LG - E400	0	0	100	0	
Nokia 800 Lumia	100	0	0	0	
Samsung - GT 5830M	0	0	0	100	
TPR by cluster	**100%**	**100%**	**100%**	**100%**	**80%**

If there are multiple clusters with the same number of photos from a device or a cluster with the same number of photos from several devices and in turn these being the highest, the cluster that is taken as predominant for the device is one chosen among the different options. It may be the case that if there is an extra cluster, a cluster may not be predominant for any device (see Table 5.2) and its TPR for this cluster is 0. Or there might be one less cluster (see Table 5.3). In this case the cluster where there are pictures of several devices, it will be associated with the device with more images have in this cluster, so this cluster will be the predominant for this device. In the case that the maximum

number of images of various devices in one cluster are equal, it will take any of them. The TPR calculation of each cluster will take into account in each cluster the number of photos of their predominant device. To calculate the final TPR in this case we must make the sum of all TPRs of each cluster and dividing by the total number of devices initially used for classifying (in the case of Table 5.3 it will be divide by 5). In Tables 5.1, 5.2 and 5.3 there are examples that illustrate the calculation of the TPR for the three cases that may occur.

In the results of the experiments 3 possible cases are considered: a) The number of identified clusters is equal to the number of devices, b) the number of identified clusters is higher than the number of devices, and c) the number of identified clusters is lower than the number of devices. Although the first case is ideal, in the second case classifications that do not mix different types of devices in a same cluster can be obtained.

We can divide the experiments according to different criteria:

- Comparison between taking the 1024×1024 region from the corner or from the center of the photograph.
- Symmetrical or asymmetrical distribution of photographs (same or different number of pictures per device).
- Comparison of grouping among devices from the same manufacturer but different model.
- Train and then carry out a classification.

5.3.1 Comparison Between Crop Corner and Crop Center

Several experiments were conducted to compare the results between cropping the image from the center or from the upper left corner, this last criterion having a TPR higher (except in the case of 7 devices with 100 images whose difference is minimal). Table 5.4 shows the TPR according to the different number of devices used and the number of photos used by device. All devices have the same number of photos.

Table 5.4: Symmetric clustering TPR by device and number of photos per device

Number of	Crop Corner			Crop Center		
Photos	3 Devices	5 Devices	7 Devices	3 Devices	5 Devices	7 Devices
50	99.33%	99.20%	99.71%	66.67%	80%	99.71%
100	74.25%	100%	87.13%	66.67%	80%	87.13%

In addition, Table 5.4 shows that TPR increases in the case of the crop in the center as more devices are grouped, whereas the crop from the corner maintains the good results for the case of 50 images per device and the results are different in the case of 100 images per device. Although [Li10a] mentions that the areas in the corners are more likely to be

saturated and therefore the noise pattern may be affected, the proposed algorithm shows the opposite in the grouping of images.

5.3.2 Symmetrical Clustering

Following is additional information about three of the symmetrical clustering experiments previously conducted whose results generate equal, lower and higher number of clusters than there are devices respectively for their classification. For each experiment, once the algorithm of clustering proposed in this work has been applied, graph will be shown for each generated cluster. This chart shows the degree of correlation of the noise pattern of all the images used in each experiment with respect to the centroid (average of all noise patterns contained in the cluster) of each cluster generated by the algorithm.

The first experiment conducts a symmetrical clustering of 3 devices, 50 images and crop corner. As shown in Table 5.5 its TPR is 99.33%. Table 5.5 shows the confusion matrix detailing generated clusters and the images included in each of them.

As can be observed, an equal number of clusters to devices is generated, which in principle is indicative of possibly obtaining a good result. The classification in this case is nearly perfect, with the exception that in cluster 2 there is an image from the E400 LG which should have been classified into cluster 3.

Table 5.5: Confusion matrix of clustering of 3 devices

Brand - Model	**Cluster 1**	**Cluster 2**	**Cluster 3**	**Average TPR**
Apple - Iphone 5	50	0	0	
Huawei - U8815	0	50	0	
LG - E400	0	1	49	
TPR by cluster	**100%**	**100%**	**98%**	**99.33%**

For this experiment Figure 5.3 to 5.5 show the 3 correlation graphs described above for each of the generated clusters.

In Figure 5.3 we can see that the correlations of a device with respect to the remaining correlations are distant, therefore a cluster with images from a single device is properly generated. In Figure 5.4 we can observe that there is an image from the E400 LG whose degree of correlation is not in line with the rest from the same device and those from the Huawei U8815. This specific image is the one which was classified incorrectly, obtaining a lower correlation degree with respect to the centroid than the remaining images of its own cluster, but different from zero and notably higher (on the order of 20 to 400 times higher correlation) to the correlation of the remaining images. Figure 5.5 shows that the pictures of the LG E400 have a similar correlation except for one which practically has a value of 0 correlation with respect to the centroid of cluster 3 and which corresponds to the erroneously classified image.

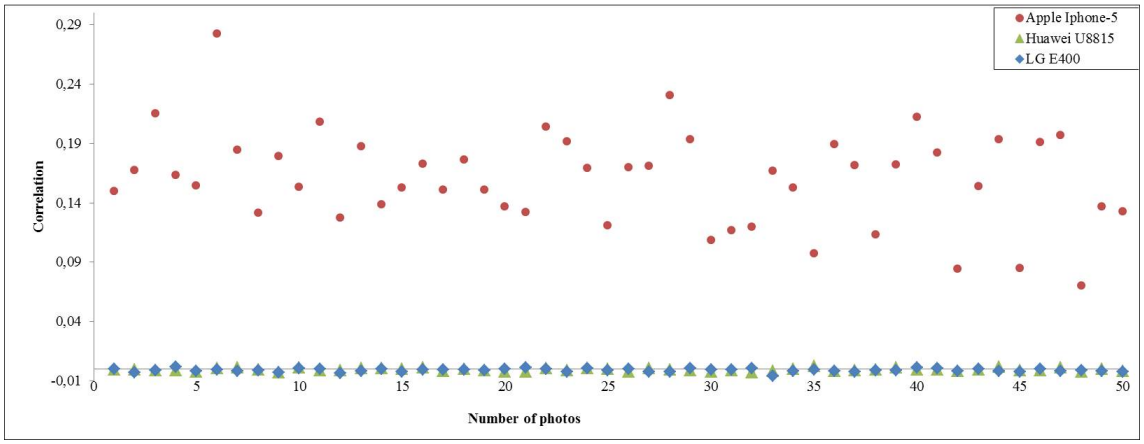


Figure 5.3: Cluster 1 correlation graphic respect to the clusters centroid

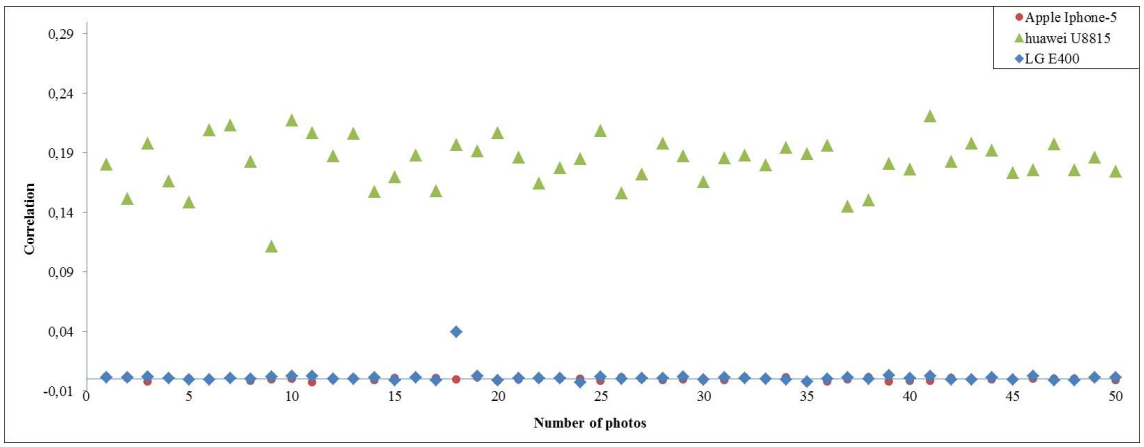


Figure 5.4: Cluster 2 correlation graphic respect to the clusters centroid

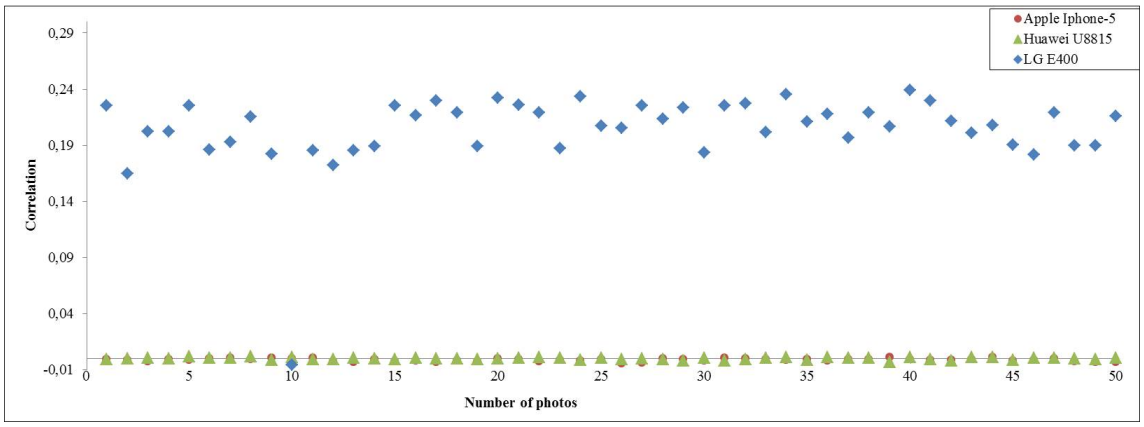


Figure 5.5: Cluster 3 correlation graphic respect to the clusters centroid

The second experiment conducts a symmetrical clustering of 5 devices, 50 images and crop center. As can be seen in Table 5.6 the TPR is 80%. Table 5.6 shows a confusion matrix detailing generated clusters and the images included in each of them.

Table 5.6: Confusion matrix of clustering of 5 devices

Brand - Model	Cluster 1	Cluster 2	Cluster 3	Cluster 4	Average TPR
Apple - Iphone 5	50	0	0	0	
Huawei - U8815	0	50	0	0	
LG - E400	0	0	50	0	
Nokia - 800 Lumia	50	0	0	0	
Samsung - GT 5830m	0	0	0	50	
TPR by cluster	**100%**	**100%**	**100%**	**100%**	**80%**

As can be observed, a lower number of clusters than of devices is generated, which implies that at least one of the clusters is not pure, i.e. it contains pictures of at least two devices. The classification in this case is completely correct for three of the four generated clusters. In contrast, all the pictures from the Apple iPhone 5 and Nokia 800 Lumia devices are in cluster 1. For this experiment Figures 5.6 to 5.9 shows the 4 correlation graphs described above for each of the generated clusters.

As shown in Figures 5.7, 5.8 and 5.9, which correspond to clusters with all images from a single device, the correlation of the images from the correctly classified device with respect to the other is distant. For these cases the correlation with respect to the centroid of the image outside the cluster is approximately zero in all cases.

Figure 5.6 shows that the correlation of images from the Apple iPhone 5 and the Huawei U8815 is similar forming cluster 1 and there is a big difference with the correlation of the rest of the images, which is close to zero.

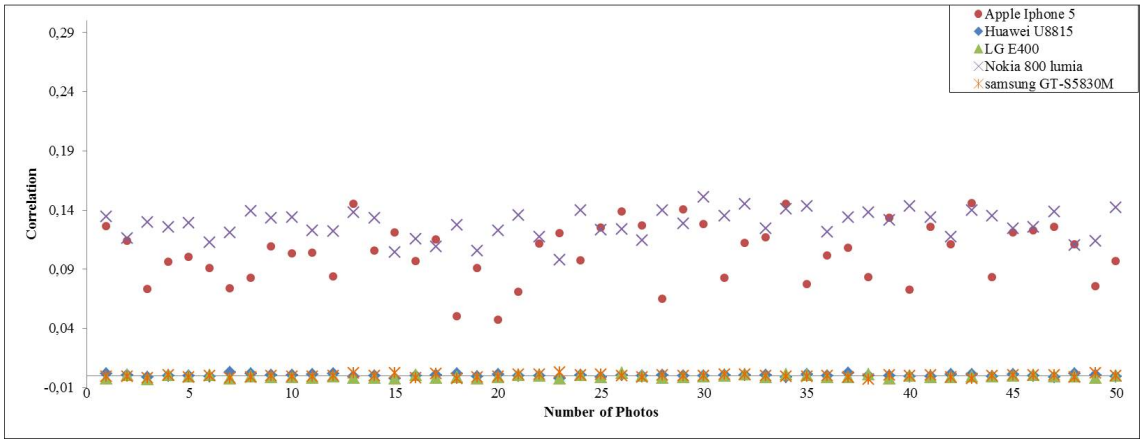


Figure 5.6: Cluster 1 correlation graphic respect to the clusters centroid

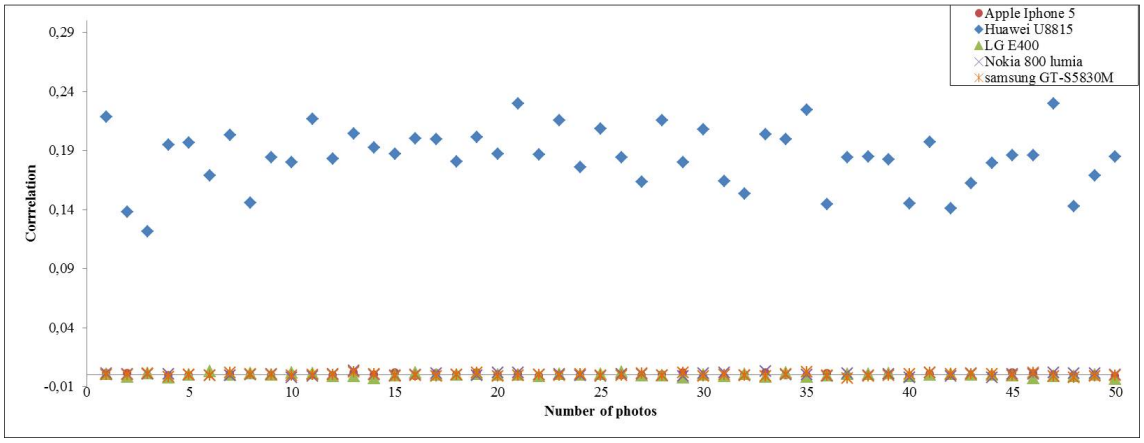


Figure 5.7: Cluster 2 correlation graphic respect to the clusters centroid

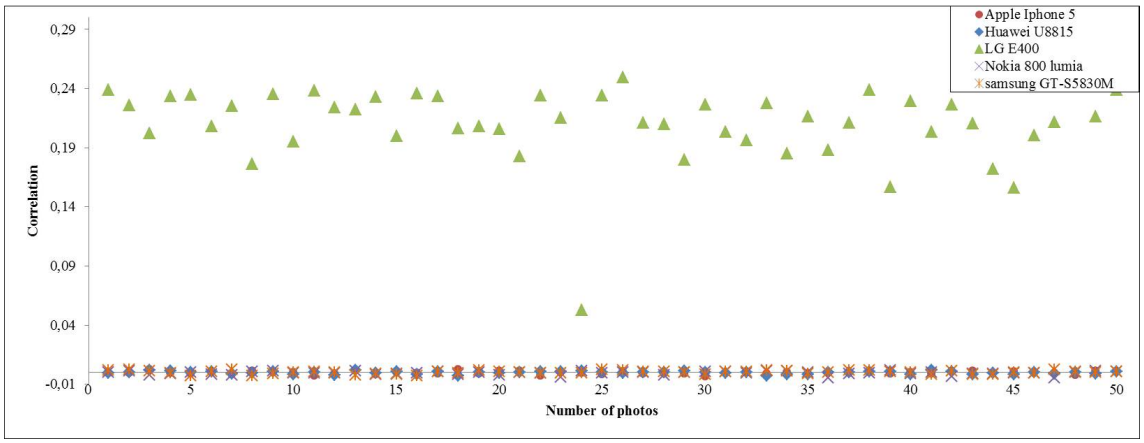


Figure 5.8: Cluster 3 correlation graphic respect to the clusters centroid

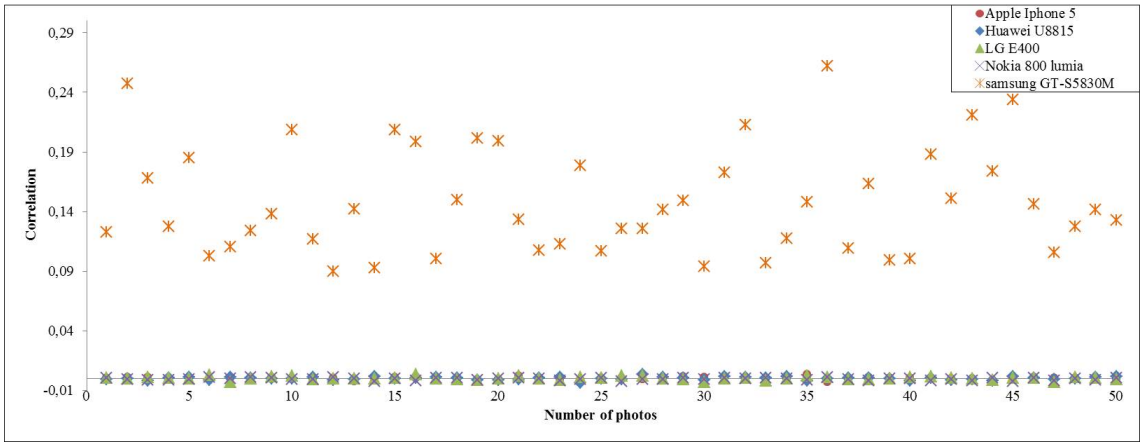


Figure 5.9: Cluster 4 correlation graphic respect to the clusters centroid

The third experiment conducts a symmetrical clustering of 7 devices, 100 images, and crop corner. As can be seen in Table 5.4, its TPR is 87.13%. Table 5.7 shows a confusion matrix detailing generated clusters and the images included in each of them.

Table 5.7: Confusion matrix of clustering of 7 devices

Brand - Model	**Clusters**								**Average TPR**
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	
Apple - Iphone 5	100	0	0	0	0	0	0	0	
Huawei - U8815	0	100	0	0	0	0	0	0	
LG - E400	0	0	97	0	0	0	0	3	
Nokia - 800 Lumia	0	0	0	100	0	0	0	0	
Samsung - GT 5830m	0	0	0	0	100	0	0	0	
Sony - ST25A	0	0	0	0	0	100	0	0	
Zopo - ZP980	0	0	0	0	0	0	100	0	
TPR by Cluster	**100%**	**100%**	**97%**	**100%**	**100%**	**100%**	**100%**	**0%**	**87.13%**

As can be observed, a higher number of clusters than of devices is generated. The classification in this case is completely correct for six of the eight generated clusters. However, clusters 3 and 8 contain images from the LG E400. While the LG E400 images were divided into two clusters, these only have images from a single device, which is a positive aspect to take into account.

For this experiment Figures 5.10 to 5.17 shows the 8 correlation graphs described above for each of the generated clusters.

As it can be seen in Figures 5.10, 5.11, 5.13, 5.14, 5.15 and 5.16, which correspond to clusters that were generated correctly, the correlation of the classified device with respect to the others is distant. For these cases the correlation with respect to the centroid of the image outside the cluster approaches zero, as in previous experiments.

In Figure 5.12 there are 97 images from the LG E400 with a correlation that is significantly higher than those of the rest of the images. There are 3 images of the LG E440 whose correlation is practically 0 and very distant from the rest of the photographs from the same device in this cluster, cluster 8 being formed by these.

In Figure 5.17, the result of the grouping of these three images on a separate cluster can be observed. The correlation of these three images is significantly higher than that of the rest of the images, which have near-zero correlations.

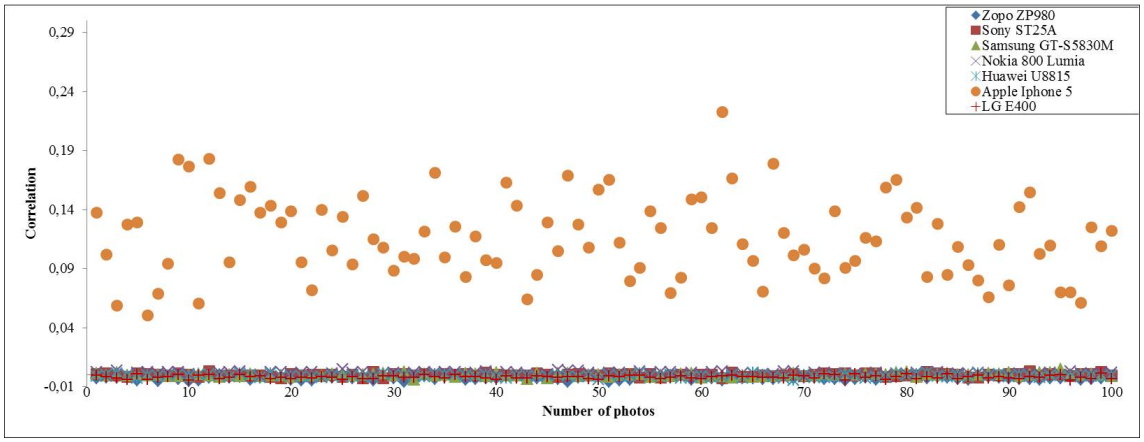


Figure 5.10: Cluster 1 correlation graphic respect to the clusters centroid

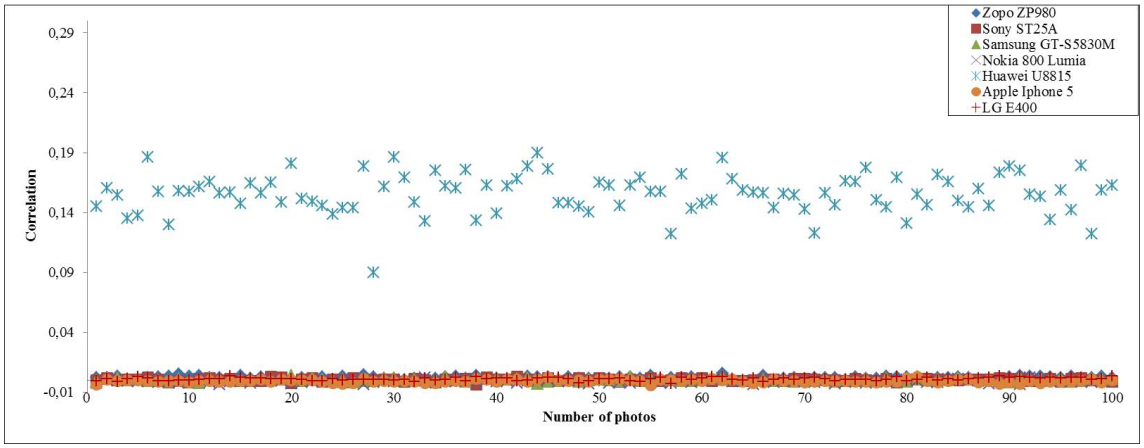


Figure 5.11: Cluster 2 correlation graphic respect to the clusters centroid

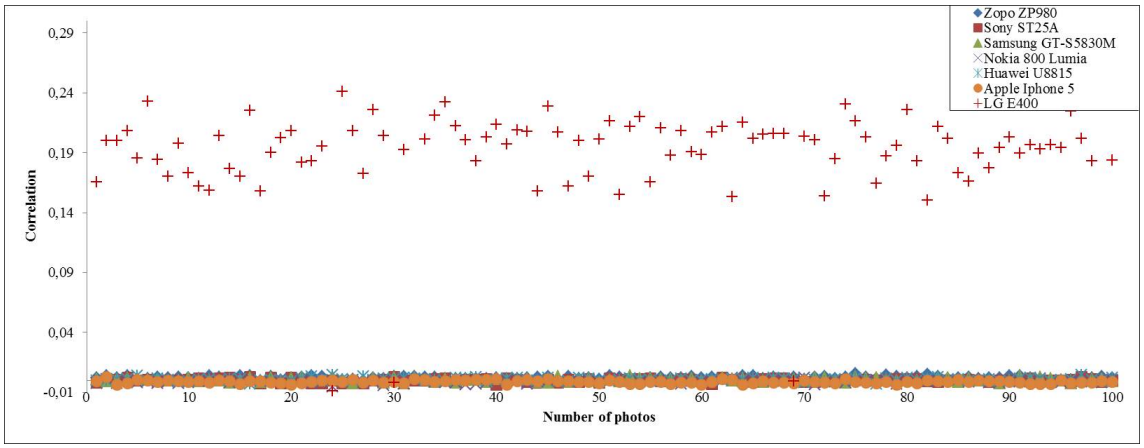


Figure 5.12: Cluster 3 correlation graphic respect to the clusters centroid

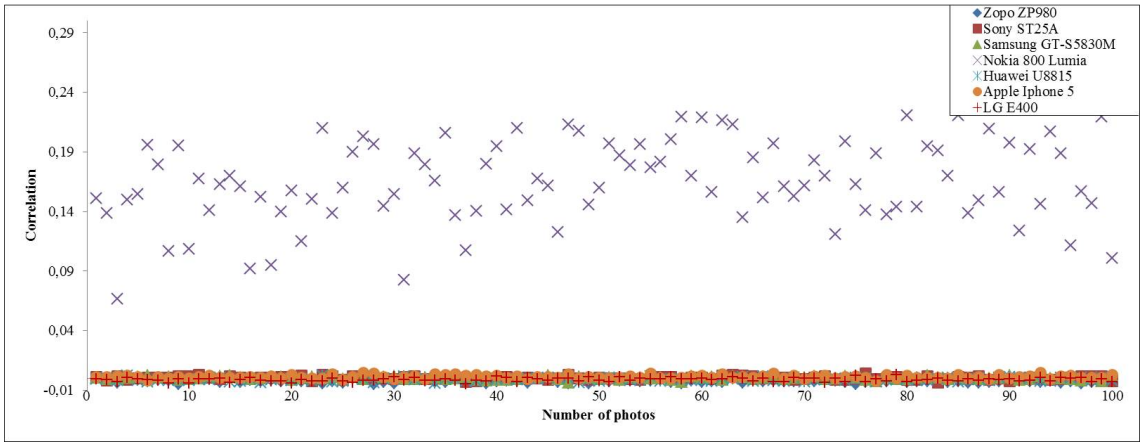


Figure 5.13: Cluster 4 correlation graphic respect to the clusters centroid

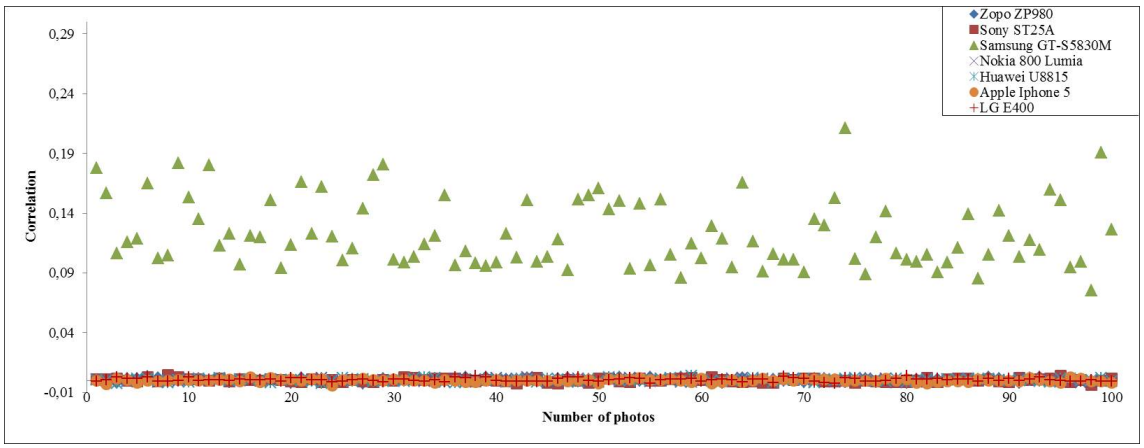


Figure 5.14: Cluster 5 correlation graphic respect to the clusters centroid

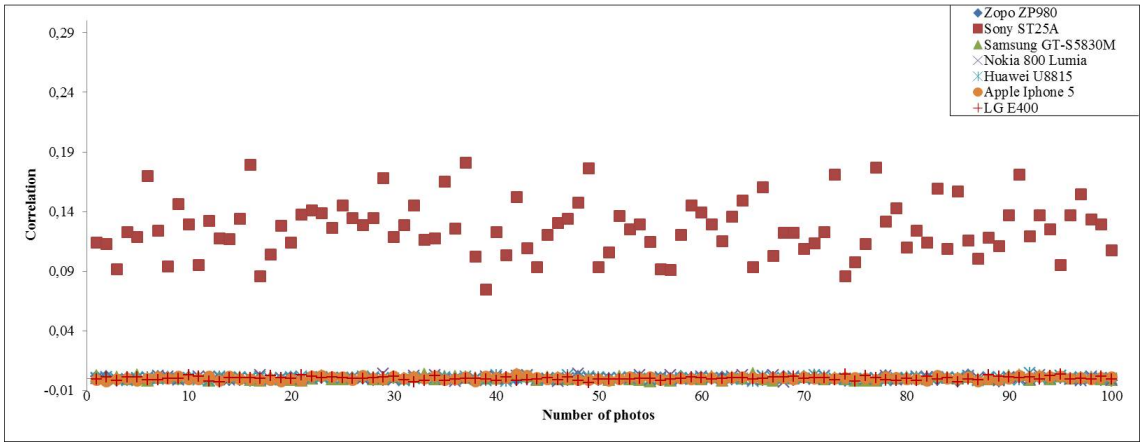


Figure 5.15: Cluster 6 correlation graphic respect to the clusters centroid

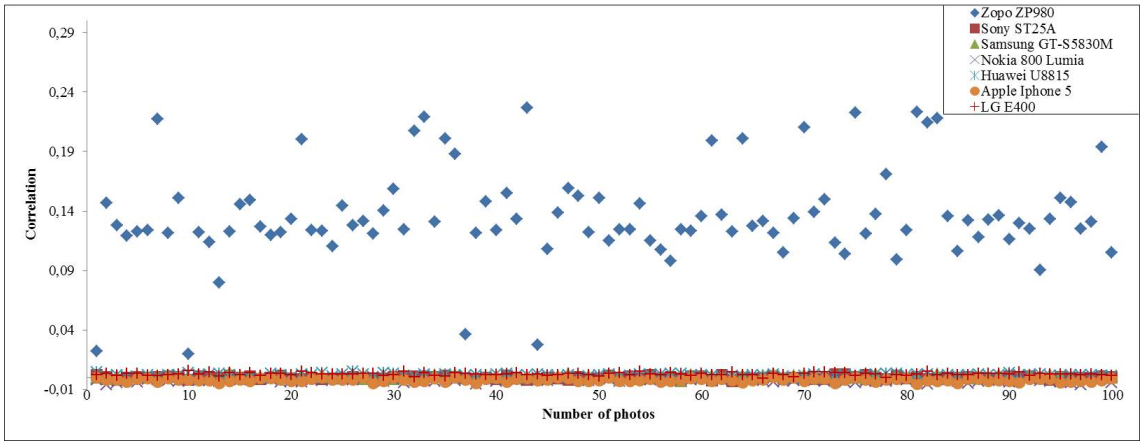


Figure 5.16: Cluster 7 correlation graphic respect to the clusters centroid

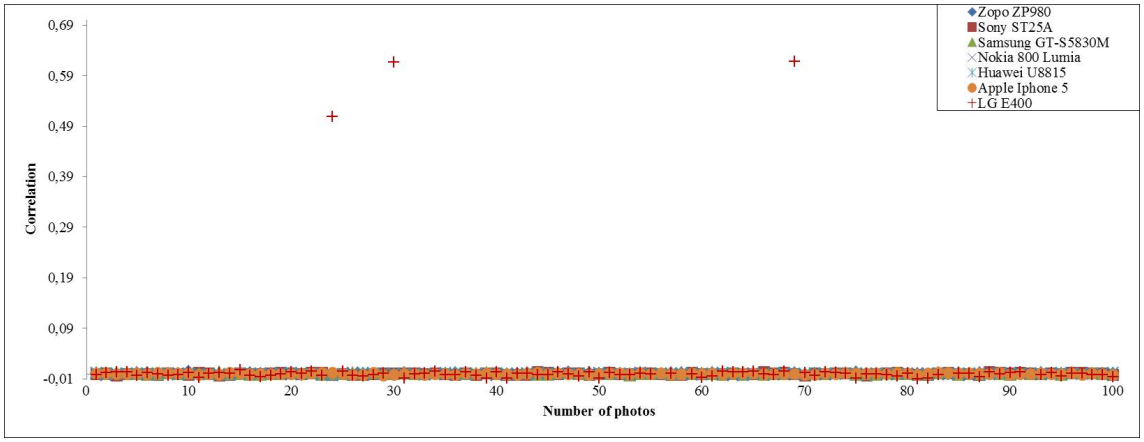


Figure 5.17: Cluster 8 correlation graphic respect to the clusters centroid

5.3.3 Asymmetrical Clustering

In a closed scenario, it is not very likely to have the same number of images from each device to identify, for that reason experiments were conducted where the sets of images for each device do not possess the algorithm proposed in a real scenario. The results of grouping images from 5 and 7 devices respectively are presented in Tables 5.8 and 5.9. The number of images per device is varied and we can still observe a high degree of success (97.28% average TPR of the experiments in Tables 5.8 and 5.9).

As can be seen, in the cases of an asymmetrical number of images there have been experiments with groups with significant numerical disparity and in some cases with small groups (5 images from a type of device), yet there have been successful grouping results.

Table 5.8: Asymmetric clustering TPR for 5 devices

Group	Apple Iphone 5	Huawei U8815	LG E400	Nokia 800 Lumia	Samsung GT 5830m	TPR
A	50	50	50	50	50	**99.20%**
B	100	100	100	100	100	**100%**
C	100	95	90	85	80	**99.78%**
D	50	45	40	35	30	**99.1%**
E	100	75	50	25	10	**99.6%**
F	100	30	20	10	5	**99%**

Table 5.9: Asymmetric clustering TPR for 7 devices

Group	Apple Iphone 5	Huawei U8815	LG E400	Nokia 800 Lumia	Samsung GT 5830m	Sony ST25a	Zopo Zp980	TPR
A	50	50	50	50	50	50	50	**99.71%**
B	100	100	100	100	100	100	100	**87.13%**
C	100	95	90	85	80	75	70	**99.84%**
D	50	45	40	35	30	25	20	**99.36%**
E	100	75	50	25	10	5	1	**85.43%**
F	100	50	40	30	20	10	5	**99.21%**

5.3.4 Same Manufacturer Different Models Clustering

Also, experiments were conducted to test the proposal in a scenario where all devices are from the same manufacturer but different models. Camera phones from the same manufacturer should be very similar for many of their products and therefore the sensor noise extracted from different models should be similar. However, Table 5.10 shows the classification TPR, concluding that for this experiment the correlation value between several SPNs varies enough among different models to identify each one separately, even when the models are very similar as is the case of the ST25a and ST25i models.

Table 5.10: Asymmetric clustering TPR for 3 devices of the same brand

Models of Sony Ericsson	TPR for 50	TPR for 100
C2105	99.33%	74.50%
ST25a		
ST25i		

5.3.5 Clustering with Training Stage

Although clustering methods do not possess information about the data sets to group, some classification experiments were conducted on a set of 5 different devices with different

sets of training both with symmetrical and asymmetrical distributions. In what refers strictly to the training stage the proposal [CAPI10] is used. This way computational complexity is widely reduced at the time of calculating the similarity matrix, this being what takes more time in the execution of the algorithm. Table 5.11, shows the number of images used for the training stage. In all experiments the remaining images were classified to 150, which the total set of images from each device possesses, no image in the training set is in the classification set and vice versa. Good results are maintained even with asymmetric image sets (98.3% of TPR for the 4 experiments).

Table 5.11: Clustering for 5 devices with training stage

Group	Apple Iphone 5	Huawei U8815	LG E400	Nokia 800 Lumia	Samsung GT 5830m	TPR
A	50	50	50	50	50	**98.67**%
B	100	100	100	100	100	**99.33%**
C	100	75	50	25	10	**96.67%**
D	50	45	40	35	30	**98.53%**

5.3.6 Clustering Algorithm Execution Times

Table 5.12 shows the clustering algorithm execution times, regardless of the features extraction, of some of the concrete experiments. Only in cases where it has made the training stage, Table 5.12 shows the total number of images used for each device detailing how many images are used for training stage and how many images are used for classification. These experiments sample times can offer a general idea of the order in runtime of the algorithm. All images were cropped to 1024×1024. The crew where the experiments were conducted is an Intel Core i7-2670QM 2.2 GHz with 6 Gb RAM and Linux operating system.

Table 5.12: Execution times

Clustering	Number of Devices	Number of Photos (Train-Classification)	Train stage	Time (sec)
Symmetric	3	50	No	445
		100	No	1698
	5	50	No	1207
		150 (100-50)	Yes	516
		100	No	4789
		150 (50-100)	Yes	1222
Asymmetric	5	50, 45, 40, 35, 30	No	742
		100, 95, 90, 85, 80	No	4299
	7	50, 45, 40, 35, 30, 25, 20	No	1484
		100, 95, 90, 85, 80, 75, 70	No	8992

5.4 Summary

This chapter has analyzed the main unsupervised image grouping techniques, which are of utmost importance in the digital image forensic analysis.

The algorithm of this proposal is based on the combination of a hierarchical clustering and a flat clustering for the separation between clusters. The use of the silhouette coefficient for cluster validation proved to report good results when obtaining high TPRs; also, the number of clusters corresponded to the number of actual devices in most cases. The percentage of correct hits when using image cropping from the left corner obtains better results than when the clipping is centered in the image, despite finding different observations in the literature arguing the saturation and lack of lighting found in those regions.

It was also important to have experimented with different device models from the same manufacturer, because along with the high rate of correct hits by using symmetric and asymmetric distributions of images by the device, it checks the adaptability of the algorithm to be applied in a real case.

Experiments conducted in this chapter have revealed a great diversity of situations with regard to the symmetry or not of the photo sets, their size, the number of devices used and the use of devices of the same brand. After all the experiments, it is concluded that the results of the application of the technique are good (92.98% TPR on average for all the experiments).

Chapter 6

A PRNU-based Counter-Forensic Method to Manipulate Smartphone Image Source Identification Techniques

Digital image forensics has become a topic of interest in recent years. Image forensic analysis arises with the idea of restoring the reliability of digital images which otherwise could be considered very easily modifiable. Just as most fields of study have a countercurrent, in this case, people like spies, criminals or scammers make efforts to manipulate images for their own benefit. They use the knowledge of digital image forensics with the aim of deleting or even supplanting the fingerprints or traces that are used to determine the image source. Many of the forensic algorithms in the literature were not designed to be robust against such behavior, and as a result, they are easy to fool. In the same way, image forensic methods may benefit from studies like us on attack techniques, with the purpose of strengthening next generation algorithms.

This chapter presents two algorithms for the destruction and falsification of digital image identity. Initially, there is a brief presentation of various attacks that can occur in the forensic analysis. The algorithm for image destruction is shown in detail below. Subsequently, the algorithm of image identity forgery is presented. Subsequently, in order to evaluate the validity of the presented algorithms, a set of experiments and the results are exposed. The chapter ends with a brief synthesis of what is presented in it.

6.1 Overview

In this section two algorithms are presented, one to destroy the image identity and another to forge a given image identity.

The aim of the first algorithm is anonymize an image, or in other words, remove as much as possible any trace that allows the source image acquisition identification by a forensic analyst. The aim of the second algorithm is forgery the source of acquisition of

an image, or in other words, doing unsuccessful a forensic analysis which has the aim of identify the image source acquisition and make its result the falsified source acquisition.

In [RCAGSO+13] it is determined that the use of the sensor pattern noise (PRNU) together with the wavelet transform is an effective method for source identification, reaching an average success rate of 87.21%. An estimated counter-forensic technique against this type of identification may be based on these elements. Therefore, the presented algorithms base their working on PRNU noise handling and wavelet transform.

6.2 Algorithm of Destruction of Image Identity

In this section an algorithm based on [GFH06] to extract and to remove the sensor fingerprint from an image P_1 is proposed. The algorithm proposed obtains a features vector for classification purposes. The proposed algorithm, on the other hand, has the aim of obtaining an image with its identity destroyed removing the PRNU noise.

Among different filters that exist for eliminating noise from images those using the wavelet transform work better because the residual noise obtained with this kind of filter contains the fewest features from the scene. Generally, the areas around the edges are misinterpreted when a less robust noise removal filter is only used, such as the Wiener filter or median filter. For this reason we selected the noise removal filter based on wavelet transform in combination with Wiener filter. For each wavelet decomposition level we obtain the high-frequency components H (horizontal), V (vertical) and D (diagonal). The Algorithm 6 describes the steps to remove the sensor fingerprint.

Where I_{clean} is obtained by applying some elimination filter as described in Section 3.3.2. Particularly in this work the noise elimination is performed by applying the Algorithm 6. Note that the I_{clean} obtained is not exactly a picture without any noise since the sensor noise pattern is formed by the PRNU and PNU as shown in Section 3.3.2. The Algorithm 6 only removes the PRNU noise of the I_{clean} and FPN does not. However for ease in the naming and nomenclature will be used I_{clean}, to name the image without PRNU noise.

6.3 Algorithm of Forgery of Image Identity

In this section we propose an algorithm to forge the sensor pattern noise from a camera C_1 to an image P_2 belonging to a camera C_2 without requiring access to camera C_2. This algorithm uses Algorithm 6 previously presented in Section 6.2.

Source identification Techniques based on PRNU estimate the sensor fingerprint of images using the equation:

$$I_{noise} = \quad I \quad - \quad I_{clean} \tag{6.1}$$

Algorithm 6: Removing the PRNU

Input: I the victim image

```
1  procedure RemovePRNU(I)
2      Apply a wavelet decomposition in 4 levels to I;
3      foreach wavelet decomposition level do
4          foreach wavelet component c ∈ {H, V, D} do
5              Compute the local variance;
6              if adaptive variance then
7                  Compute 4 variances with windows
                    of size: 3, 5, 7 and 9 respectively;
8                  Select the minimum variance;
               else
9                  Compute the variance with a window
                    of size 3;
10         Compute noiseless wavelet components
            applying the Wiener filter to the variance;
11     Obtain I_clean by applying the inverse wavelet transform with clean components
       calculated;
12 end procedure
```

The pattern noise PNU is computed by averaging the residual noise of several images with the following equation:

$$P_{noise} = \frac{1}{N} \sum_{i=1}^{N} I_{noise} \tag{6.2}$$

Once it is possible to remove sensor noise and to extract the sensor pattern noise, the image identity falsification could be envisaged. The algorithm 7 shows the steps to follow to fake the image identity.

Algorithm 7: Forging the PRNU

Input: I the victim image
N the number of flat images from forger camera

```
1  procedure ForgeImg(I, N)
2      I_clean ← RemovePRNU(I);
3      P_noise ← ExtractPRNU(N);
4      Apply a wavelet decomposition in 1 levels to
        I_clean obtaining components L_I, H_I, V_I and D_I;
5      Apply a wavelet decomposition in 1 levels to
        P_noise obtaining components H_P, V_P and D_P;
6      Compute the forged wavelet components
        with c_F=c_I+c_P where c∈{H,V,D};
7      Obtain I_forged applying the inverse wavelet
        transform with L_I,H_F,V_F y D_F;
8  end procedure
```

In order to have a better quality pattern and to obtain better results in forgery a number of images N bigger than 50, are recommended according to experiments, also the images have been acquired from non-textured uniformly lighted flat surfaces, as flat surfaces could be considered pictures of clear sky or white paper.

6.4 Experiments

This section describes the experiments performed with the algorithms for removing the sensor fingerprint (Algorithm 6) and for forging the source camera fingerprint (algorithm 7).

The experiments on elimination and falsification of sensor fingerprints were performed using the proposed implementations and the tool "PRNU Decompare" [Net13b], which uses the flatfielding technique described in Section 3.3.2 and allows the elimination and falsification of sensor pattern noise. This tool requires a picture of a dark frame as input and a number N of images of flat surfaces uniformly lighted (a minimum of 30 frames is recommended). We compared the results of our proposal algorithms with the obtained results in the experiments. For this purpose we used the tool "NFI PRNU Compare" [Net13a], this tool allows us to compare images and sensor pattern noises from various images. "NFI PRNU Compare" uses as a measure to compare the correlation, which is employed in many other works such as [GFC11], [CESR12] and [UH12] among others, to compare images and noise patterns. It is important to note that our proposal does not need a set of images from the camera with the identity we want to destroy from the picture, we only need the image itself. Also, we do not need any set of photos or have access to the victim camera in the case of forgery of the identity, we only need a set of pictures of the attacker camera. That is, our proposed algorithm requires less input images than "PRNU Decompare" to do the same function.

6.4.1 Destruction of Image Identity

For this experiment photos from 6 digital cameras (LG E510f, LG 400, Nokia 800 Lumia, Sony ST25i (Xperia U), Apple iPhone5, Samsung GTI-9000) were used. From each device 50 photos of uniformly lighted flat images were taken, and one totally dark picture was generated by completely covering the camera lens. One picture was randomly selected from the photo database of each camera to be used for the removal of the image identifiable data. All photos were cropped to a size of 1024×1024 pixels.

Initially, a first set of images without fingerprint was generated using the Algorithm 6. It should be noted that the noise elimination only required the picture which sought to eliminate the sensor noise and not additional ones. Then, a second group of images without fingerprint was generated using the tool "PRNU Decompare", using 50 flat images and a dark frame image as input to this program. Therefore we have two images without sensor camera fingerprint for each camera, one generated with the Algorithm 6 and one with "PRNU Decompare".

For evaluating the effectiveness of the algorithm for elimination of sensor fingerprint six sets of images were compared using the tool "NFI PRNU Compare". With 40 images

chosen randomly from the 50 of each camera, "NFI PRNU Compare" gets the noise pattern of each camera, as it is indicated in the recommendations of the tool documentation.

Table 6.1 shows the results of comparing each sensor pattern noise generated by "NFI PRNU Compare" of each camera with the noiseless images generated by the two tools, and also against the original photograph. "NFI PRNU Compare" allows us to measure how far the patterns compared are similar to each other, the rows closer to the pattern that is being compared are the most similar to it. Also, Table 6.1 shows the correlation coefficients for each color channel. The closer the value to 1 is considered a high correlation with a high degree of similarity between two linear patterns, a value of 0.5 represents a weak correlation and a negative value indicates a negative correlation in which the increase of a value involves decrease the other. In all tests the noiseless images generated with "PRNU Decompare" resulted to be least similar to the pattern, this was expected as they considered much more information to remove the fingerprint. In all the cases, the comparison of the noiseless images had very similar results, indicating that in this case the proposed algorithm had a good performance getting close to the result of "PRNU Decompare" without the need to use a dark frame nor the 50 flat images.

Table 6.1: Comparison between patterns and noiseless images

Pattern	Image	Red	Green	Blue	Sum
LG E510f	Original	-0.014645672	-0.0017777978	-0.007864626	-0.024288096
	Proposal	-0.015506644	-0.003044259	-0.008411303	-0.026962206
	Decompare	-0.018929206	-0.0023383496	-0.012027217	-0.033294775
LG E400	Original	0.011481647	0.010190065	0.01825918	0.039930895
	Proposal	0.010315191	0.008225861	0.017940063	0.036481116
	Decompare	0.010638827	0.009045472	0.016430777	0.036115076
Nokia 800 Lumia	Original	0.011352311	0.011754888	0.019119238	0.042226437
	Proposal	0.009912337	0.009113852	0.016991276	0.036017465
	Decompare	0.010812875	0.009995133	0.014978331	0.035786339
Apple iPhone 5	Original	-0.015712192	-0.002311284	-0.007307031	-0.025330507
	Proposal	-0.016984729	-0.003462913	-0.009599754	-0.030047396
	Decompare	-0.019395503	-0.003140395	-0.009915681	-0.032451579
Sony ST25i	Original	-0.012013772	-0.002127295	-0.006817536	-0.020958603
	Proposal	-0.014839721	-0.003142763	-0.009114359	-0.027096843
	Decompare	-0.016545112	-0.002611324	-0.011200112	-0.030356548
Samsung GTI-9000	Original	0.017310016	0.010754888	0.016119238	0.044184142
	Proposal	0.014015311	0.007826601	0.014491394	0.036333306
	Decompare	0.014979864	0.007992510	0.012984029	0.035956403

An important issue to address is whether the image whose identity has been destroyed reduces its image quality or if effects visible to the human eye exist on the image scene. For it, in Figure 6.1 two examples of images of Nokia 800 Lumia and Sony ST25i (Xperia U) devices are shown with their respective destroyed identity images.

(a) Nokia 800 Lumia: Original Image

(b) Nokia 800 Lumia: Destructed identity image

(c) Sony ST25i: Original Image

(d) Sony ST25i: Destructed identity image

Figure 6.1: Images and destructed identity images

As it can be seen image scenes whose identity has been destroyed do not reveal visual changes.

Even so, it was decided to use IQM to objectively assess the loss of image quality whose identity has been destroyed with respect to the original.

For this, it was decided to use the Minkowski metrics [HLZ10]. In particular we use the Minkowsky metric for $\gamma = 1$ which corresponds to MAE and $\gamma = 2$ with the MSE. In both cases, high values of MAE or MSE correspond with low quality images. These metrics are applied to each of the RGB bands separately, so that three metrics for the MAE and another three for the MSE are obtained. The values of the quality metrics for each of the bands of the 4 pictures are shown in Table 6.2, as it can be seen the

destruction of image identifiable attributes changes very little in the image quality indexes thus presented. Although a very small reduction is perceived nearly everywhere, it is important to note that in reality the original images will likely be not available, and the changes are so minimal that will not allow to distinguish original or forged images solely based on them.

Table 6.2: MAE and MSE of the images and destructed identity images

Image	MAE			MSE		
	Red	Green	Blue	Red	Green	Blue
Nokia 800 Lumia original	0.8410	0.8183	0.8428	2.6498	2.1777	2.3869
Nokia 800 Lumia destructed identity	0.8368	0.8135	0.8388	2.6245	2.1531	2.3617
Sony ST25i original	0.8976	0.8705	0.8916	4.0664	3.3754	3.7108
Sony ST25i destructed identity	0.8924	0.8656	0.8869	4.0324	3.3465	3.6783

6.4.2 Forgery of Image Identity

For this experiment the same picture set of the experiment 6.4.1 was used. In a similar way to the previous experiment, a camera sensor fingerprint was extracted, then this fingerprint was injected into pictures from another two cameras using the proposed algorithm and "PRNU Decompare", and finally the results were compared with "NFI PRNU Compare". The roles played by each camera are shown in Table 6.3.

Table 6.3: Devices used for identity forgery

Forger Camera	Victim 1	Victim 2
LG E510f	LG-400	Samsung GT-I8160P
Forger Camera	**Victim 3**	**Victim 4**
Nokia 800 Lumia	Sony ST25i	Samsung GTI-9000

To forge the sensor pattern noise with the proposed algorithm only 50 images from the forger camera were required. For the falsification with "PRNU Decompare" 50 pictures were required as well as one dark frame belonging to the forger and to the victim camera, respectively. After performing forgery in 4 victim cameras the results were compared as summarized in Table 6.4. Also, this Table shows correlation coefficients for each color channel with respect to the pattern of the forger camera as in the last experiment.

Table 6.4: Comparison of patterns, original images and victims

Image		Red	Green	Blue	Sum
Victim 1	Decompare	0.009219962	0.0054620425	0.009098741	0.023780745
	Proposal	0.007900867	0.0046529872	0.0083521	0.020905953
	Original	0.0073570074	0.004183661	0.0075896666	0.019130334
Victim 2	Decompare	0.01418404	0.013986045	0.013574668	0.041744754
	Original	0.011300047	0.013949845	0.0125216115	0.0377715
	Proposal	0.008964902	0.0066337977	0.004440412	0.020039111
Victim 3	Decompare	0.00811001	0.004442147	0.009333811	0.021885968
	Proposal	0.007201833	0.00419231	0.009112659	0.020506802
	Original	0.006232409	0.003880702	0.007871138	0.017984249
Victim 4	Decompare	0.009913582	0.007213382	0.008788015	0.025914979
	Proposal	0.008271773	0.007621801	0.008141919	0.024035493
	Original	0.008137003	0.005338104	0.00790911	0.021384217

For victims 1, 3 and 4 it can be observed that the three forgeries have a greater similarity with the pattern of the forger camera and that the result of "PRNU Decompare" is closer than the proposal, even though this difference is not significant considering that they use a far greater number of images as a source of information. In the case of victim 2 the result with the proposed algorithm has the least similarity to the forged camera pattern.

The results obtained so far were as expected, because the algorithm proposed does not assume having access to the victim camera as the tool "PRNU Decompare" does. However, in all the cases the results of our proposal are close to the results of "PRNU Decompare". It is important to notice that in real scenarios, it is not normally possible to access the victim camera. We just need a set of photos of the attacker camera, as "PRNU Decompare" needs. However we do not need any special type of content scenery of the set of images and "PRNU Decompare" needs it, as we discussed above.

As in the case of destruction of image identity, the same image quality loss study for the forgery example of victims 3 and 4 is performed. The results can be seen visually in Figure 6.2, observing that changes are imperceptible to the human eye.

(a) Victim 3 original

(b) Victim 3 forged identity

(c) Victim 4 original

(d) Victim 4 forged identity

Figure 6.2: Images and forged identity images

In Table 6.5 image quality indexes of MAE and MSE of the original images and their respective forged ones are shown.

As can be seen in Table 6.5 the forgery of an image identifiable data does not change significantly any of the image quality indexes presented, and this difference will be almost impossible to notice if the original images where not provided.

Table 6.5: MAE and MSE of the images and forged identity images

Image	MAE			MSE		
	Red	Green	Blue	Red	Green	Blue
Victim 3 original	0.8976	0.8705	0.8916	4.0664	3.3754	3.7108
Victim 3 forged	0.8926	0.8656	0.8872	4.0323	3.3445	3.6787
Victim 4 original	0.7836	0.7800	0.7820	2.3412	2.2525	2.2724
Victim 4 forged	0.7685	0.7639	0.7661	2.2940	2.2062	2.2207

6.5 Summary

In this chapter, two new algorithms were proposed, one capable of removing all identifiable data from an image, and another that allows its forgery. The two algorithms are based on the use of different types of sensor noise and wavelet transform. Both algorithms, in addition, have the great advantage over existing ones that they need a much smaller and easier to obtain data input, which make them more applicable and closer real-life scenarios.

Both algorithms can be viewed as research contributions to the future strengthening of forensic techniques for detecting intentional manipulation. For example, our first algorithm can be very useful in web applications that upload and display images on the Internet (social networks, directories, etc.), as they allow for complete anonymity when uploading an image.

The effectiveness of the proposed algorithms is overall quite good, even though in some cases they do not get results comparable with those of other algorithms, but they generally achieve close and quite acceptable results, with the benefit of drastically reducing the required input data and being more practical and realistic. These algorithms can be useful as a starting point for future improvements that allow similar results to be obtained by other algorithms or tools, emphasizing and taking into account the limited input data needed and that they can work even when not having access to the victim's camera in the case of identity image forgery. Also, the application of both algorithms does not cause noticeable visual changes, or degradation in the pictures and does not significantly reduce the image quality.

Chapter 7

Conclusions and Future Work

Large numbers of digital images are circulating daily on the Internet or are used as evidence or proof in judicial proceedings. As a consequence, forensic analysis of digital images generated by devices such as digital camera, mobile device, scanner or computer becomes important in many real-life situations. It is noteworthy that specific image forensic techniques are required for mobile devices, and aren't valid in most cases because there are significant intrinsic features that differentiate these types of devices. An example of this is presented in a situation where a forensic analyst needs to identify the type (camera, scanner, computer) or class (make and model) of the image acquisition source.

In this study, various techniques for identifying mobile device images with respect to scanned and computer-generated images are presented. Besides, other techniques that allow us to distinguish the acquisition source of smartphone images are presented. The techniques are based on the use of four feature sets (Noise, Color, IQM and Wavelets), on which adjustments have been made in order to improve the results for this specific type of devices. There have been experiments with the combination of the different feature sets, different crop sizes and positions, and wavelet functions. The classification is done with SVMs.

With regard to source type identification, the first general conclusion is that noise features are discarded as invalid when the number of types of devices is greater than 2. This is because in the experiments unacceptable results were obtained while identifying among three device types (scanner, mobile, and computer). In the experiments that used whole images and different crop sizes and positions, unacceptable results were obtained for identifying three types of devices (scanner, mobile, and computer). As discussed in the experiments, for these three types of devices there are dozens of different manufacturers and models, hampering classification.

As a counterpart, forensic analysts may consider the application of the technique with noise features for identifying the source type of images from mobile devices with respect to images from scanners and computers. The results are quite good at identifying the type when discerning between scanners and mobile devices. The use of all the features significantly improves results, but as a general conclusion, they are not good enough to be used in a compromising situation.

When identifying the acquisition source of mobile device images, the results are much

more encouraging. In all sets of experiments performed, there is at least one configuration that yields better results, always putting them in the context of the level of demand on this technique (a large number of devices or many devices from the same manufacturer).

The use of all feature sets or wavelets from Daubechies 8-tap family, both are the ones yielding better results. Regarding crop size, there is an optimal size for obtaining optimal results, which does not necessarily have to be the largest or the whole image, since the latter produces worse results than when using a crop. When taking a sufficiently large crop size, for example, 1024×768, reducing the crop lowers the success rate. Regarding the number of devices used, as expected, the larger number of devices the lower the success rate. The same holds true when devices from the same manufacturer, whose cameras are similar or identical in some cases, are used.

Therefore, the forensic analyst, knowing a priori information in some cases, and taking these conclusions into account, must decide on setting the various parameters of the technique and the validity of the results, taking into account the percentages obtained in the experiments presented in this work. That is to say, a single application of the technique can yield good results in some cases and bad results in others, depending on factors such as whether or not we want to decide the type or make and model of the devices, the number of devices used or the number of devices by the same manufacturer, amongst other things.

The source identification techniques using SVMs as classifiers where it is necessary to know a priori the classes to which the images belong in some scenarios are not applicable, for example situations where the analyst does not know the set of cameras to which a set of images may belong. In this way, other techniques based on clustering arise, in which there are no a priori camera data and the objective is not to identify camera make and model, but to be able to group different images in disjoint groups in which all their images belong to the same device.

This study has made an analysis of the main unsupervised image grouping techniques, which are of utmost importance in digital image forensic analysis. Despite the rise of mobile device cameras these days, there are still few references to unsupervised mobile device image grouping. Most of the works refer to the supervised classification and in many cases; they do not focus on mobile device images, which have unique characteristics.

The results from this work cannot be accurately compared to other results as they do not refer to the final number of clusters generated, which is a fundamental issue. Moreover, in these works there are no details given on how the success rates were calculated nor is there any reference made to them when the clusters generated by the classification are different in number to the number of devices used, making the comparison of their rates with respect to our interpretation of the *True Positive Rate* (TPR) meaningless.

The noise added to each image by the camera sensor due to faults in the manufacturing process or defects by daily use has proved to be a trustworthy source of source identification. Likewise, the normalized correlation calculation between sensor noises extracted from two or more images is a quite used similarity measure in image unsupervised learning techniques, being the clustering techniques those with better results.

The algorithm of this proposal is based on the combination of a hierarchical clustering

and a flat clustering for the separation between clusters. The use of the silhouette coefficient for cluster validation proved to report good results when obtaining high TPRs; also, the number of clusters corresponded to the number of actual devices in most cases. The percentage of correct hits when using image cropping from the left corner obtains better results than when the clipping is centered in the image, despite finding different observations in the literature arguing the saturation and lack of lighting found in those regions.

It was also important to have experimented with different device models from the same manufacturer, because along with the high rate of correct hits by using symmetric and asymmetric distributions of images by device, it checks the adaptability of the algorithm to be applied in a real case.

Experiments conducted in this work have revealed a great diversity of situations with regard to the symmetry or not of the photo sets, their size, the number of devices used and the use of devices of the same brand. After all the experiments, it is concluded that the results of the application of the technique are good (92.98% TPR on average for all the experiments).

It is important to take into account the different attacks that proposed techniques could suffer in order to improve them. Accordingly, two algorithms have been presented, one capable of removing all identifiable data from an image, and another that allows its forgery. The two algorithms are based on the use of different types of sensor noise and wavelet transform.

Both algorithms, in addition, have the great advantage over existing ones that they need a much smaller and easier to obtain data input, which make them more applicable and closer real-life scenarios. In particular, the removal algorithm requires only the image from which to wipe the fingerprint, not a set of planar images and a dark frame from the camera as in others approaches.

The algorithms that require a large set of photos with specific characteristics for its proper functioning are unrealistic in most cases, since the victim's camera can hardly be accessed. In the case of the proposed image forgery algorithm, it does not need to access the victim's camera.

Both algorithms can be viewed as research contributions to the future strengthening of forensic techniques for detecting intentional manipulation. For example, the first algorithm can be very useful in web applications that upload and display images on the Internet (social networks, directories, etc.), as they allow for completely anonymous image uploading from the standpoint of acquisition source identification.

The effectiveness of the proposed algorithms is overall quite good, even though in some cases they do not get results comparable with those of other algorithms, but they generally achieve close and quite acceptable results, with the benefit of drastically reducing the required input data and being more practical and realistic.

These algorithms can be useful as a starting point for future improvements that allow similar results to be obtained by other algorithms or tools, emphasizing and taking into account the limited input data needed and that they can work even when not having access to the victim's camera in the case of identity image forgery. Also, the application of

both algorithms does not cause noticeable visual changes, or degradation in the pictures, and does not significantly reduce the image quality.

7.1 Future Work

The information from these conclusions can be a starting point for future work, among which the following can be pointed out:

- **Search for new sets of characteristics for image acquisition source identification**: Research on new features and configurations, either to improve on their own or in conjunction with those presented in this work the success rates or to present new alternatives for different applications. Particularly in cases where a success rate very close to 100% is needed as in applications related to court cases. In the same way when the number of devices is very large, in order to be able to apply the algorithms to extensive forensic analysis databases.
- **Improve clustering techniques results**: The main objective of this improvement is to maximize the number of classes generated by the technique, that is, in all or in the vast majority of cases, the number of devices to be classified matches to the number of sets created by the clustering algorithm. Once this goal has been met, the second order would be to optimize the group's homogeneity, avoiding generated classes by the technique, which contain objects from different devices.
- **Clustering based on image features**: The use of other image features different from those of SPN for classifying images, but continuing using the same clustering algorithm could improve or give new applications to clustering. For this purpose, we will study wavelets, color and IQM features.
- **Clustering for videos**: This aim will be to apply these clustering techniques to digital videos generated by mobile devices.
- **Improve forensic analysis techniques robustness**: Improve presented and future techniques to be stronger with respect to different possible attacks. This aspect has not been considered during the creation of different techniques, as the main objective is to have techniques with good results for images that have not been manipulated. Once this goal is achieved, the techniques should be strengthened to make them more robust against attacks.
- **Image identity destruction optimization**: The image identity destruction should not be considered merely as an attack. Images uploaded on the web in many cases may require this function. Consequently, it's important to study how to eliminate all or most of any trace type that allows image identification, ensuring that image quality is not lost.

Bibliography

[AG11] D. M. Arenas González. Análisis Forense de Imágenes de Móviles mediante el Uso de Metadatos. MSc Thesis, Facultad de Informática, Universidad Complutense de Madrid, Spain, September 2011.

[AG15] D. M. Arenas González. Técnicas de Identificación de la Fuente de Adquisición en Imágenes Digitales de Dispositivos Móviles. PhD Thesis, Facultad de Informática, Universidad Complutense de Madrid, Spain, April 2015.

[AGRCSO+14] D. M. Arenas González, J. Rosales Corripio, A. L. Sandoval Orozco, J. A. Zapata Guridi, and L. J. García Villalba. Clasificación sin Supervisión de Imágenes de Dispositivos Móviles. In *Actas de la XIII Reunión Española sobre Criptología y Seguridad de la Información*, pages 271–276, Alicante, Spain, September 2014.

[ALE16] Alexa Top 500 Global Sites. `http://www.alexa.com/topsites`, December 2016.

[AM12] T. Ahonen and A. Moore. Tomi Ahonen Almanac 2012: Mobile Telecoms Industry Annual Review. `http://www.tomiahonen.com/ebook/almanac.html`, May 2012.

[AM14] T. Ahonen and A. Moore. Tomi Ahonen Almanac 2014: Mobile Telecoms Industry Annual Review. `http://communities-dominate.blogs.com/`, July 2014.

[AMS03] I. Avcibas, N. Memon, and B. Sankur. Steganalysis Using Image Quality Metrics. *IEEE Transactions on Image Processing*, 12(2):221–229, April 2003.

[APS98] J. Adams, K. Parulski, and K. Spaulding. Color Processing in Digital Cameras. *IEEE Micro*, 18(6):20–30, December 1998.

[ASS02] I. Avcibas, B. Sankur, and K. Sayood. Statistical Evaluation of Image Quality Measures. *Journal of Electronic Imaging*, 11(2):206–223, April 2002.

[AZ06] M. Al-Zarouni. Mobile Handset Forensic Evidence: a Challenge for Law Enforcement. In *Proceedings of the 4th Australian Digital Forensics Conference*, pages 1–10, Perth Western, Australia, December 2006.

[Bae10] R. Baer. Resolution Limits in Digital Photography: The Looming End of the Pixel Wars - OSA Technical Digest (CD). In *Proceedings of the Imaging Systems Conference*, page ITuB3, Tucson, Arizona, USA, June 2010.

[BK13] R. Bohme and M. Kirchner. Counter-Forensics: Attacking Image Forensics. In H. T. Sencar and N. Memon, editors, *Digital Image Forensics*, pages 327–366. Springer New York, July 2013.

[BL04] M. Boutell and J. Luo. Photo Classification by Integrating Image Content and Camera Metadata. In *Proceedings of the 17th International Conference on Pattern Recognition*, volume 4, pages 901–904, Cambridge, United Kingdom, August 2004.

[BL05] M. Boutell and J. Luo. Beyond Pixels: Exploiting Camera Metadata for Photo Classification. *Pattern Recognition*, 38(6):935–946, June 2005.

[Blo08] G. J. Bloy. Blind Camera Fingerprinting and Image Clustering. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30(3):532–535, January 2008.

[BM13] G. K. Birajdar and V. H. Mankar. Digital Image Forgery Detection Using Passive Techniques: A Survey. *Digital Investigation*, 10(3):226–245, October 2013.

[BR93] J. D. Banfield and A. E. Raftery. Model-Based Gaussian and Non-Gaussian Clustering. *Biometrics*, 49(3):803–821, September 1993.

[BSM06] S. Bayram, H. T. Sencar, and N. Memon. Improvements on Source Camera-Model Identification Based on CFA Interpolation. In *Proceedings of the International Conference on Digital Forensics*, pages 24–27, Orlando, Florida, USA, February 2006.

[BSM08] S. Bayram, H. T. Sencar, and N. Memon. Classification of Digital Camera-Models Based on Demosaicing Artifacts. *Digital Investigation*, 5(1-2):49–59, September 2008.

[CAPI10] R. Caldelli, I. Amerini, F. Picchioni, and M. Innocenti. Fast Image Clustering of Unknown Source Images. In *Proceedings of the IEEE International Workshop on Information Forensics and Security*, pages 1–5, Seattle, Washington, USA, December 2010.

[CAS+06] O. Celiktutan, I. Avcibas, B. Sankur, N. P. Ayerden, and C. Capar. Source Cell-Phone Identification. In *Proceedings of the 14th IEEE Signal Processing and Communications Applications*, pages 1–3, Antalya, Turkey, April 2006.

[CESR12] F. D. O. Costa, M. Eckmann, W. J. Scheirer, and A. Rocha. Open Set Source Camera Attribution. In *Proceedings of the 25th Conference on Graphics, Patterns and Images*, pages 71–78, Ouro Preto, Brazil, August 2012.

[CFGL08] M. Chen, J. Fridrich, M. Goljan, and J. Lukas. Determining Image Origin and Integrity Using Sensor Noise. *IEEE Transactions on Information Forensics and Security*, 3(1):74–90, March 2008.

[Cho06] K. S. Choi. Source Camera Identification Using Footprints from Lens Aberration. In *Proceedings of the Conference on Digital Photography II*, number 852 in 6069, pages 60690J–60690J–8, San Jose, California, USA, February 2006.

[CIS16] Cisco Visual Networking Index: Global Mobile Data Traffic Forecast Update, 2016-2021 White Paper. `goo.gl/ELWfMM`, February 2016.

[CK10] H. Cao and A. C. Kot. Mobile Camera Identification Using Demosaicing Features. In *Proceedings of 2010 IEEE International Symposium on Circuits and Systems*, pages 1683–1686, Paris, France, May 2010.

[CL13] C. C. Chang and C. J. Lin. LIBSVM: A Library for Support Vector Machines. Version 3.17. `http://www.csie.ntu.edu.tw/~cjlin/libsvm/`, April 2013.

[CSS07] W. Chen, Y.Q. Shi, and W. Su. Image Splicing Detection Using 2-D Phase Congruency and Statistical Moments of Characteristic Function. In *Proceedings of the SPIE Conference on Security, Steganography, and Watermarking of Multimedia Contents*, volume 6505, pages 65050R–65050R–8, San Jose, California, USA, January 2007.

[dOCSE+14] F. de O. Costa, E. Silva, M. Eckmann, W. J. Scheirer, and A. Rocha. Open Set Source Camera Attribution and Device Linking. *Pattern Recognition Letters*, 39(0):92–101, February 2014.

[DSM14] A. E. Dirik, H. T. Sencar, and N. Memon. Analysis of Seam-Carving-Based Anonymization of Images Against PRNU Noise Pattern-Based Source Attribution. *IEEE Transactions on Information Forensics and Security*, 9(12):2277–2290, December 2014.

[EZZ04] C. F. Eick, N. Zeidat, and Z. Zhao. Supervised Clustering-Algorithms and Benefits. In *Proceedings of the IEEE International Conference on Tools with Artificial Intelligence*, pages 774–776, Boca Raton, Florida, USA, November 2004.

[FBA97] T. Frese, C. A. Bouman, and J. P. Allebach. Methodology for Designing Image Similarity Metrics Based on Human Visual System Models. In *Proceedings of the SPIE /IS&T Conference on Human Vision and Electronic Imaging II*, volume 3016, pages 472–483, San Jose, California, USA, June 1997.

[Fis87] D. H. Fisher. Knowledge Acquisition Via Incremental Conceptual Clustering. *Machine Learning*, 2(2):139–172, September 1987.

[FKCS11] J. Fan, A. C. Kot, H. Cao, and F. Sattar. Modeling the EXIF-Image Correlation for Image Manipulation Detection. In *Proceedings of the 18th IEEE International Conference on Image Processing*, pages 1945–1948, Brussels, Belgium, October 2011.

[Fri09] J. Fridrich. Digital Image Forensics. *IEEE Signal Processing Magazine*, 26(2):26–37, March 2009.

[GBK+01] Z. J. Geradts, J. Bijhold, M. Kieft, K. Kurosawa, K. Kuroki, and N. Saitoh. Methods for Identification of Images Acquired with Digital Cameras. In *Proceedings on Enabling Technologies for Law Enforcement and Security*, volume 4232, pages 505–512, Boston, Massachusetts, USA, February 2001.

[GFC11] M. Goljan, J. Fridrich, and Mo Chen. Defending Against Fingerprint-Copy Attack in Sensor-Based Camera Identification. *IEEE Transactions on Information Forensics and Security*, 6(1):227–236, March 2011.

[GFF09] M. Goljan, J. Fridrich, and T. Filler. Large Scale Test of Sensor Fingerprint Camera Identification. In *Proceedings on Media Forensics and Security*, volume 7254, pages 72540I–72540I, San Jose, California, USA, February 2009.

[GFH06] M. Goljan, J. Fridrich, and T Holotyak. New Blind Steganalysis and its Implications. In *Proceedings of the SPIE, Security, Steganography, and Watermarking of Multimedia Contents VIII*, volume 6072, pages 607201–607201–13, San Jose, California, USA, January 2006.

[GKWB07] T. Gloe, M. Kirchner, A. Winkler, and R. Bohme. Can We Trust Digital Image Forensics? In *Proceedings of the 15th International Conference on Multimedia*, pages 78–86, Augsburg, Germany, September 2007.

[GVSORC15] L. J. García Villalba, A. L. Sandoval Orozco, and J. Rosales Corripio. Smartphone Image Clustering. *Expert Systems with Applications*, 42(4):1927–1940, March 2015.

[GVSORCHC] L. J. García Villalba, A. L. Sandoval Orozco, J. Rosales Corripio, and J. C. Hernández Castro. A PRNU-Based Counter-Forensic Method to Manipulate Smartphone Image Source Identification Techniques (In Press). *Future Generation Computer Systems*. DOI:10.1016/j.future.2016.11.007.

[Ham92] E. Hamilton. JPEG File Interchange Format. Version 1.02. `http://www.w3.org/Graphics/JPEG/jfif3.pdf`, September 1992.

[HAZG10] J. S. Ho, O. C. Au, J. Zhou, and Y. Guo. Inter-channel Demosaicking Traces for Digital Image Forensics. In *Proceedings of the IEEE International Conference on Multimedia and Expo*, pages 1475–1480, Singapore, July 2010.

[HCHY15] W.-C. Hu, W.-H. Chen, D.-Y. Huang, and C.-Y. Yang. Effective Image Forgery Detection of Tampered Foreground or Background Image Based on Image Watermarking and Alpha Mattes. *Multimedia Tools and Applications*, 75(6):3495–3516, January 2015.

[HCL03] C. W. Hsuand, C. C. Chang, and C. J. Lin. A Practical Guide to Support Vector Classification. Practical Guide, Department of Computer Science and Information Engineering, National Taiwan University, April 2003.

[HKKR99] F. Hoppner, F. Klawonn, R. Kruse, and T. Runkler. *Fuzzy Cluster Analysis: Methods for Classification, Data Analysis and Image Recognition*. Wiley, July 1999.

[HLZ10] Y. Hu, C.-T. Li, and C. Zhou. Selecting Forensic Features for Robust Source Camera Identification. In *Proceedings of the International Computer Symposium*, pages 506–511, Tainan, Taiwan, December 2010.

[JB14] S. Jothimani and P. Betty. Image Authentication using Global and Local Features. In *Proceedings of the International Conference on Green Computing Communication and Electrical Engineering*, pages 1–5, Coimbatore, India, March 2014.

[JLLC07] C.-J. Jang, J.-Y. Lee, J.-W. Lee, and H.-G. Cho. Smart Management System for Digital Photographs Using Temporal and Spatial Features with EXIF Metadata. In *Proceedings of the Second International Conference on Digital Information Management*, volume 1, pages 110–115, Lyon, France, October 2007.

[KD15] A. Karakucuk and A. E. Dirik. Adaptive Photo-Response Non-Uniformity Noise Removal Against Image Source Attribution. *Digital Investigation*, 12:66–76, 2015.

[KDSM15] A. Karakucuk, A. E. Dirik, H. T. Sencar, and N. D. Memon. Recent Advances in Counter PRNU Based Source Attribution and Beyond. In *Proceedings of the Media Watermarking, Security, and Forensics Conference*, volume 9409, pages 94090N–94090N–11, San Francisco, California, USA, February 2015.

[KMC+07] N. Khanna, A. K. Mikkilineni, G. T. Chiu, J. P. Allebach, and E. J. Delp. Forensic Classification of Imaging Sensor Types. In *Proceedings of the SPIE, Security, Steganography, and Watermarking of Multimedia Contents IX*, volume 6505, pages 65050U–65050U–9, San Jose, California, USA, February 2007.

[KMD09] N. Khanna, A. K. Mikkilineni, and E. J. Delp. Scanner Identification Using Feature-Based Processing and Analysis. *IEEE Transactions on Information Forensics and Security*, 4(1):123–139, January 2009.

[LF03] J. Lukas and J. Fridrich. Estimation of Primary Quantization Matrix in Double Compressed JPEG Images. In *Proceedings of the Digital Forensic Research Workshop*, pages 5–8, Cleveland, Ohio, USA, August 2003.

[LFG06] J. Lukas, J. Fridrich, and M. Goljan. Digital Camera Identification from Sensor Pattern Noise. *IEEE Transactions on Information Forensics and Security*, 1(2):205–214, June 2006.

[LH06] Y. Long and Y. Huang. Image Based Source Camera Identification Using Demosaicking. In *Proceedings of the IEEE 8th Workshop on Multimedia Signal Processing*, pages 419–424, Victoria, British Columbia, Canada, October 2006.

[Li10a] C.-T. Li. Source Camera Identification Using Enhanced Sensor Pattern Noise. *IEEE Transactions on Information Forensics and Security*, 5(2):280–287, June 2010.

[Li10b] C.-T. Li. Unsupervised Classification of Digital Images Using Enhanced Sensor Pattern Noise. In *Proceedings of the IEEE International Symposium on Circuits and Systems*, pages 3429–3432, Paris, France, May 2010.

[LLC+12] Q. Liu, X. Li, L. Chen, H. Cho, A. P. Cooper, Z. Chen, M. Qiao, and A. H. Sung. Identification of Smartphone-Image Source and Manipulation. In He Jiang, Wei Ding, Moonis Ali, and Xindong Wu, editors, *Advanced Research in Applied Artificial Intelligence*, volume 7345 of *Lecture Notes in Computer Science*, pages 262–271. Springer Berlin Heidelberg, Dalian, China, June 2012.

[LLHC10] B.-B. Liu, H.-K. Lee, Y. Hu, and C.-H. Choi. On Classification of Source Cameras: A Graph Based Approach. In *Proceedings of the IEEE International Workshop on Information Forensics and Security*, pages 1–5, Seattle, Washington, USA, December 2010.

[LP05] R. Lukac and K. N. Plataniotis. Color Filter Arrays: Design and Performance Analysis. *IEEE Transactions on Consumer Electronics*, 51(4):1260–1267, November 2005.

[LWLD13] G. Liu, J. Wang, S. Lian, and Y. Dai. Detect Image Splicing with Artificial Blurred Boundary. *Mathematical and Computer Modelling*, 57(11):2647–2659, June 2013.

[MSGW08] C. Mckay, A. Swaminathan, H. Gou, and M. Wu. Image Acquisition Forensics: Forensic Analysis to Identify Imaging Source. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 1657–1660, Las Vegas, Nevada, USA, June 2008.

[MST94] D. Michie, D. J. Spiegelhalter, and C. C. Taylor. *Machine Learning, Neural and Statistical Classification*. Ellis Horwood, February 1994.

[Nak05] J. Nakamura. *Image Sensors and Signal Processing for Digital Still Cameras*. CRC Press, Boca Raton, Florida, USA, August 2005.

[NB92] B. N. Nill and B. Bouzas. Objective Image Quality Measure Derived from Digital Image Power Spectra. *Optical Engineering*, 31(4):813–825, April 1992.

[Net13a] Netherlands Forensic Institute. NFI PRNU Compare, May 2013.

[Net13b] Netherlands Forensic Institute. PRNU Decompare, May 2013.

[Nil85] N. Nill. A Visual Model Weighted Cosine Transform for Image Compression and Quality Assessment. *IEEE Transactions on Communications*, 33(6):551–557, June 1985.

[OA11] L. Ozparlak and I. Avcibas. Differentiating Between Images Using Wavelet-Based Transforms: A Comparative Study. *IEEE Transactions on Information Forensics and Security*, 6(4):1418–1431, December 2011.

[Oht07] J. Ohta. *Smart CMOS Image Sensors and Applications*. CRC Press, September 2007.

[PF05] A. C. Popescu and H. Farid. Exposing Digital Forgeries by Detecting Traces of Resampling. *IEEE Transactions on Signal Processing*, 53(2):758–767, February 2005.

[Pla00] J. Platt. AutoAlbum: Clustering Digital Photographs Using Probabilistic Model Merging. In *Proceedings of the IEEE Workshop on Content-Based Access of Image and Video Libraries*, pages 96–100, Hilton Head Island, South Carolina, USA, June 2000.

[RCAGSO+13] J. Rosales Corripio, D. M. Arenas González, A. L. Sandoval Orozco, L. J. García Villalba, J. C. Hernández Castro, and S. J. Gibson. Source Smartphone Identification Using Sensor Pattern Noise and Wavelet Transform. In *Proceedings of the 5th International Conference on Imaging for Crime Detection and Prevention*, pages 1–6, London, United Kingdom, December 2013.

[RCAGSOGV14] J. Rosales Corripio, D. M. Arenas González, A. L. Sandoval Orozco, and L. J. García Villalba. Identificación de la Fuente de Imágenes de Dispositivos Móviles Basada en el Ruido del Sensor. In *Actas de la XIII Reunión Española sobre Criptología y Seguridad de la Información*, pages 277–280, Alicante, Spain, September 2014.

[RCAVSOGV16] J. Rosales Corripio, E. A. Armas Vega, A. L. Sandoval Orozco, and L. J. García Villalba. Uso de Características en la Identificación de la Fuente de Imágenes de Dispositivos Móviles. In *Actas de la XIII Reunión Española sobre Criptología y Seguridad de la Información*, pages 232–237, Mahón, Spain, October 2016.

[RCC+08] N. L. Romero, V. G. Chornet, J. S. Cobos, A. S. Carot, F. C. Centellas, and M. C. Mendez. Recovery of Descriptive Information in Images from Digital Libraries by Means of EXIF Metadata. *Library Hi Tech*, 26(2):302–315, June 2008.

[RCEKSOGV15] J. Rosales Corripio, A. El-Khattabi, A. L. Sandoval Orozco, and L. J. García Villalba. Método Anti-Forense para Manipular la Fuente de Adquisición de una Imagen de Dispositivo Móvil. In *Actas del VIII Congreso Iberoamericano de Seguridad Informática*, pages 176–182, Quito, Ecuador, November 2015.

[RCEKSOGV16] J. Rosales Corripio, A. El-Khattabi, A. L. Sandoval Orozco, and L. J. García Villalba. Theia: Una Herramienta para el Análisis Forense de Imágenes Digitales de Dispositivos Móviles. In *Actas de la XIII Reunión Española sobre Criptología y Seguridad de la Información*, pages 226–231, Mahón, Spain, October 2016.

[RCSOGV15a] J. Rosales Corripio, A. L. Sandoval Orozco, and L. J. García Villalba. New Technique of Forensic Analysis for Digital Cameras in Mobile Devices. In *Proceedings of the 7th International Conference on Information Technology*, pages 597–602, Amman, Jordan, May 2015.

[RCSOGV15b] J. Rosales Corripio, A. L. Sandoval Orozco, and L. J. García Villalba. Unsupervised Classification of Mobile Device Images. In *Proceedings of the 7th International Conference on Information Technology*, pages 96–101, Amman, Jordan, May 2015.

[Rok10] L. Rokach. A Survey of Clustering Algorithms. In Oded Maimon and Lior Rokach, editors, *Data Mining and Knowledge Discovery Handbook*, pages 269–298. Springer USA, July 2010.

[RSBG11] A. Rocha, W. Scheirer, T. Boult, and S. Goldenstein. Vision of the Unseen: Current Trends and Challenges in Digital Image and Video Forensics. *ACM Computing Surveys*, 43(4):26:1–26:42, October 2011.

[RSYD05] R. Ramanath, W. E. Snyder, Y. Yoo, and M. S. Drew. Color Image Processing Pipeline. *IEEE Transactions on Signal Processing*, 22(1):34–43, January 2005.

[SA91] S. Z. Selim and K. Alsultan. A Simulated Annealing Algorithm for the Clustering Problem. *Pattern Recognition*, 24(10):1003–1008, October 1991.

[SLFK10] M. Steinebach, H. Liu, P. Fan, and S. Katzenbeisser. Cell Phone Camera Ballistics: Attacks and Countermeasures. In *Proceedings of the Conference on Multimedia on Mobile Devices*, pages 75420B–75420B–9, San Jose, California, USA, January 2010.

[SOAGGVHC12] A. L. Sandoval Orozco, D. M. Arenas González, L. J. García Villalba, and J. C. Hernández Castro. Anomalías en el Seguimiento de Exif en el Análisis Forense de Metadatos de Imágenes de Móviles. In *Actas de la XII Reunión Española sobre Criptología y Seguridad de la Información*, Donostia-San Sebastián, Spain, September 2012.

[SOAGGVHC14] A. L. Sandoval Orozco, D. M. Arenas González, L. J. García Villalba, and J. C. Hernández Castro. Analysis of Errors in Exif Metadata on Mobile Devices. *Multimedia Tools and Applications*, 74:4735–4763, January 2014.

[SOAGRC+13] A. L. Sandoval Orozco, D. M. Arenas González, J. Rosales Corripio, L. J. García Villalba, and J. C. Hernández Castro. Techniques for Source Camera Identification. In *Proceedings of the 6th International Conference on Information Technology*, pages 1–9, Amman, Jordan, May 2013.

[SOAGRC+14] A. L. Sandoval Orozco, D. M. Arenas González, J. Rosales Corripio, L. J. García Villalba, and J. C. Hernández Castro. Source Identification for Mobile Devices, Based on Wavelet Transforms Combined with Sensor Imperfections. *Computing*, 96(9):829–841, September 2014.

[SOGVAG+15] A. L. Sandoval Orozco, L. J. García Villalba, D. M. Arenas González, J. Rosales Corripio, J. C. Hernández Castro, and S. Gibson. Smartphone Image Acquisition Forensics Using Sensor Fingerprint. *IET Computer Vision*, 9(5):723–731, October 2015.

[SORCGVAG16] A. L. Sandoval Orozco, J. Rosales Corripio, L. J. García Villalba, and D. M. Arenas González. Theia: A Tool for the Forensic Analysis of Mobile Devices Pictures. *Computing*, 98(12):1251–1286, December 2016.

[SORCGVHC16] A. L. Sandoval Orozco, J. Rosales Corripio, L. J. García Villalba, and J. C. Hernández Castro. Image Source Acquisition Identification of Mobile Devices Based on the Use of Features. *Multimedia Tools and Applications*, 75(12):7087–7111, June 2016.

[Sta10] Standarization Committee. Exchangeable Image File for Digital Still Cameras: Exif Version 2.3, April 2010.

[SWL09] A. Swaminathan, M. Wu, and K. J. R. Liu. Component Forensics. *IEEE Signal Processing Magazine*, 26(2):38–48, March 2009.

[Tes05] J. Tesic. Metadata Practices for Consumer Photos. *IEEE Multimedia*, 12(3):86–92, September 2005.

[TLL07] M. J. Tsai, C. L. Lai, and J. Liu. Camera/Mobile Phone Source Identification for Digital Forensics. In *Proceedings of the International Conference on Acoustics Speech and Signal Processing*, pages II–221–II–224, Honolulu, Hawaii, USA, April 2007.

[TNC10] V. L. L. Thing, K. Y. Ng, and E. C. Chang. Live Memory Forensics of Mobile Phones. *Digital Investigation*, 7:74–82, August 2010.

[UH12] A. Uhl and Y. Holler. Iris-Sensor Authentication Using Camera PRNU Fingerprints. In *Proceedings of the 5th IAPR International Conference on Biometrics*, pages 230–237, New Delhi, India, April 2012.

[VA00] J. Vesanto and E. Alhoniemi. Clustering of the Self-Organizing Map. *IEEE Transactions on Neural Networks*, 11(3):586–600, May 2000.

[VEK07] L. T. Van, S. Emmanuel, and M. S. Kankanhalli. Identifying Source Cell Phone Using Chromatic Aberration. In *Proceedings of the IEEE International Conference on Multimedia and Expo*, pages 883–886, Beijing, China, July 2007.

[VLCEK07] T. Van Lanh, K. S. Chong, S. Emmanuel, and M. S. Kankanhalli. A Survey on Digital Camera Image Forensic Methods. In *Proceedings of the IEEE International Conference on Multimedia and Expo*, pages 16–19, Beijing,China, July 2007.

[VTT13] G. Valenzise, M. Tagliasacchi, and S. Tubaro. Revealing the Traces of JPEG Compression Anti-Forensics. *IEEE Transactions on Information Forensics and Security*, 8(2):335–349, February 2013.

[Wat93] A. B. Watson, editor. *Digital Images and Human Vision*. MIT Press, Cambridge, Massachusetts, USA, 1993.

[WGKM09] B. Wang, Y. Guo, X. Kong, and F. Meng. Source Camera Identification Forensics Based on Wavelet Features. In *Proceedings of the International Conference on Intelligent Information Hiding and Multimedia Signal Processing*, pages 702–705, Kyoto, Japan, September 2009.

[XW05] R. Xu and D. Wunsch. Survey of Clustering Algorithms. *IEEE Transactions on Neural Networks*, 16(3):645–678, May 2005.

[Zah71] C. T. Zahn. Graph-Theoretical Methods for Detecting and Describing Gestalt Clusters. *IEEE Transactions on Computers*, C-20(1):68–86, January 1971.

Part II

Descripción de la Investigación

Capítulo 8

Introducción

Con frecuencia las fotografías son consideradas como una parte de la verdad al ser hechos reales capturados por dispositivos electrónicos (cámaras). Sin embargo, con el desarrollo de la tecnología han surgido herramientas potentes y sofisticadas que facilitan de una manera impresionante la alteración de las imágenes digitales, incluso para quienes no tienen conocimientos técnicos o especializados en el área [GKWB07].

El desarrollo de las tecnologías digitales ha estado y continúa avanzando a un ritmo imparable. Cada día el número de cámaras digitales va creciendo, así como la facilidad de acceso a ellas. Desde el año 2000, cuando el primer teléfono móvil con cámara fue introducido en el mercado, el número de usuarios de móviles se ha quintuplicado. Para el año 2020 habrá 5.500 millones de usuarios de móviles, representando el 70 % de la población mundial estimada para ese año (Figura 8.1). La proliferación de teléfonos móviles, incluyendo tabletas, está creciendo tan aceleradamente que será mayor el número de personas que poseerán móviles (5.400 millones) que las que tendrán electricidad (5.300 millones), agua potable (3.500 millones) o automóviles (2.800 millones) en 2020 [CIS16].

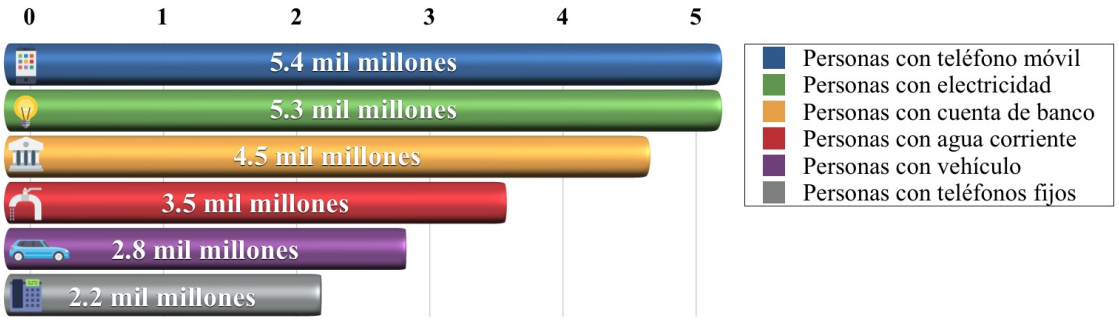


Figura 8.1: Crecimiento del número de dispositivos móviles hasta el 2020

Actualmente, más del 90 % de los dispositivos móviles cuentan con cámaras digitales integradas, los cuales, a diferencia de las cámaras digitales convencionales, abreviadamente DSCs, son llevadas por sus dueños todo el tiempo a la mayoría de lugares que asisten

y, en muchos casos, estos dispositivos tienen conexión a Internet [AM14]. Incluso existen predicciones que indican que las DSCs desaparecerán en pro de las nuevas cámaras digitales integradas en dispositivos móviles [Bae10], ya que el aumento de calidad de éstas crece a un ritmo imparable.

Debido al incremento en sus capacidades de almacenamiento, de procesamiento, de usabilidad y de portabilidad así como a su bajo coste, los dispositivos móviles están presentes en diversidad de actividades, lugares y eventos de la vida diaria. Algunos datos que permiten hacerse una idea de la magnitud de la presencia de este tipo de dispositivos son los siguientes [AM12]:

- Más del 90 % de las personas que han tomado una fotografía alguna vez lo han hecho únicamente con cámaras de dispositivos móviles.
- Un gran número de personas tienen y usan más de un dispositivo móvil.
- Las estadísticas globales arrojan que un usuario típico en promedio mira su móvil 150 veces al día y 8 de ellas es para hacer uso de la funcionalidad de la cámara.

El amplio uso de cámaras digitales de dispositivos móviles es una realidad en la vida cotidiana. Diariamente pueden verse imágenes y vídeos generados por dispositivos móviles en telenoticias, distintas aplicaciones, correo electrónico o en redes sociales. Webs como Youtube, Facebook, Instagram o Twitter, entre otras, se sitúan en los puestos más altos de la lista de webs más visitadas, siendo una parte considerable de su contenido capturado con cámaras digitales de dispositivos móviles [ALE16]. Es por esto que se han generado polémicas, discusiones y normas sobre la prohibición del uso de dispositivos móviles con cámara digital en lugares como escuelas, oficinas de gobierno, eventos empresariales, conciertos, empresas, etc.

Las imágenes digitales en la actualidad son utilizadas como testigos silenciosos en procesos judiciales, siendo una pieza crucial de la evidencia del crimen [AZ06]. De manera análoga a la balística que trata de relacionar una pistola con sus balas, el análisis forense de imágenes digitales trata de relacionar una imagen con la cámara digital con la que fue generada [WGKM09]. Es por ello que el análisis forense de imágenes digitales cobra importancia ya que podría ayudar a combatir la pornografía infantil, el robo de tarjetas de crédito, la piratería, los secuestros, el espionaje industrial, etc.

En particular, este trabajo se enfoca en los dispositivos móviles, ya que además de su extenso uso, éstos poseen características específicas que permiten obtener mejores resultados, no siendo válidas muchas veces las técnicas forenses para imágenes digitales generadas por otros tipos de dispositivos. En [TNC10] se describe detalladamente y de forma clara la necesidad de técnicas de análisis forense específicas para dispositivos móviles.

8.1 Identificación del Problema

Las imágenes digitales generadas con dispositivos móviles son archivos digitales almacenados en un soporte y codificados en un determinado formato. En la mayoría de los casos estos archivos contienen una información adicional al propio contenido visual de la imagen denominada "metadatos".

Los metadatos pueden ser eliminados o modificados con gran facilidad, ya sea o no de forma malintencionada. Por tanto, puede perderse fácilmente la pista del dispositivo que generó la imagen, los parámetros de geolocalización, la fecha, las condiciones de la captura, etc.

Debido a la vulnerabilidad de los metadatos es necesario ir más allá diseñando técnicas y algoritmos que utilicen el contenido de la imagen. Al igual que ocurre con los metadatos, las imágenes pueden ser modificadas, malintencionadamente o no, evitando que las técnicas forenses se puedan aplicar. Sin embargo, las técnicas basadas en el contenido de la imagen son más robustas, requiriéndose de un mayor nivel de conocimientos para contrarrestar el análisis que se les realiza.

Estas situaciones pueden generar problemas o indefiniciones cuando las imágenes son utilizadas como evidencias en algún proceso, ya sea judicial o no, dado que no se puede garantizar la identificación de la fuente de adquisición de la imagen o la integridad de la misma sin realizar un análisis forense previo.

La mayoría de las técnicas de identificación de la fuente de adquisición están orientadas a escenarios cerrados. Estas técnicas asumen que se tiene conocimiento a priori de la marca y modelo de los dispositivos móviles a los que podrían pertenecer las imágenes bajo análisis. No obstante, existe la necesidad de migrar a escenarios abiertos para acercarse más a la realidad, como sucede con las técnicas de agrupamiento, donde cualquier imagen puede pertenecer a cualquier dispositivo móvil.

El estudio de las técnicas anti-forenses gana valor al permitir ir un paso más adelante, y así tener la posibilidad de generar nuevas técnicas forenses o mejorar las existentes.

8.2 Motivación

Para realizar el análisis de una imagen digital generada por una cámara se utilizan los rasgos que se impregnan en ella durante su proceso de generación en el dispositivo que la crea.

Una motivación para este trabajo es que la mayoría de las investigaciones realizadas en los últimos años sobre técnicas de análisis forense de imágenes digitales se han enfocado únicamente en las imágenes generadas por cámaras tradicionales DSCs. Considerando que hoy en día las cámaras de los dispositivos móviles prácticamente han sustituido a las DSCs, surge la necesidad de enfocar las técnicas de análisis forenses de imágenes hacia los dispositivos móviles.

Además, en la literatura se detecta la necesidad de generar experimentos de identificación y agrupamiento de la fuente con un mayor número de cámaras, representando así escenarios más realistas.

En el área de las técnicas anti-forenses la mayoría de propuestas para eliminar o falsificar la fuente de una imagen asumen que tienen acceso a la cámara víctima. Surge, pues, la motivación para generar técnicas que únicamente requieran de la cámara atacante, acercándose así más a la realidad.

8.3 Objetivos

Esta investigación tiene cinco objetivos principales:

1. Desarrollar una técnica de identificación de la fuente de una imagen digital basada en las características propias de la imagen. La técnica debe ser capaz de identificar entre distintos tipos de dispositivos como computadores, escáneres o cámaras digitales de dispositivos móviles, así como entre diferentes marcas y modelos para el caso de los dispositivos móviles.

2. Crear una técnica de identificación de la fuente de imágenes digitales generadas por dispositivos móviles para escenarios abiertos, esto es, que permita realizar agrupamiento de las mismas en clases, sin entrenamiento o clases conocidas a priori.

3. Diseñar una técnica de anonimización de la identidad de una imagen digital generada por un dispositivo móvil, permitiendo de esta forma evitar ser víctima de posibles ataques.

4. Diseñar una técnica de falsificación de la identidad de una imagen digital generada por un dispositivo móvil en la que no se asuma que se tiene acceso a la cámara digital víctima.

5. Integración de las técnicas anteriores en Theia, la herramienta de análisis forense de imágenes desarrollada en el Grupo GASS.

Los objetivos 1 y 2 se enmarcan en el área de la identificación de la fuente de adquisición de imágenes, los objetivos 3 y 4 pertenecen al área de las contramedidas forenses y el objetivo 5 está relacionado con estas dos áreas del análisis forense al implementar e integrar las técnicas propuestas en una aplicación de software.

8.4 Resumen de las Contribuciones

Los resultados de la investigación realizada en esta tesis comprenden diversas contribuciones que han sido publicadas en revistas internacionales indexadas en el JCR y en congresos especializados del área. Como se representa en la Figura 8.2, estas contribuciones se enmarcan en el área del análisis forense de imágenes digitales de dispositivos móviles.

Respecto a la identificación de la fuente de adquisición de imágenes digitales, que es la primer rama del análisis forense que se aborda en esta tesis, se propone una solución al problema de la identificación en escenarios cerrados basada en las características propias de la imagen y se presenta una técnica de agrupamiento,

basada en la combinación de agrupamiento jerárquico y plano y en el uso del patrón de ruido del sensor, para llevar a cabo la identificación en escenarios abiertos. Estas dos técnicas forenses de identificación de la fuente de adquisición de imágenes generadas por dispositivos móviles se encuentran organizadas en dos capítulos: el algoritmo basado en las características de la imagen se especifica en el Capítulo 11 [SOAGRC+13] [RCSOGV15a] [SORCGVHC16] [RCAVSOGV16] y el algoritmo de agrupamiento de imágenes digitales se describe en el Capítulo 12 [RCAGSO+13] [GVSORC15] [RCAGSOGV14] [RCSOGV15b] [SOGVAG+15]. Estas técnicas de identificación se han incluido en una herramienta para el análisis forense de imágenes digitales de dispositivos móviles llamada *Theia* [RCEKSOGV16] [SORCGVAG16].

En el área de ataques al análisis se proponen dos técnicas anti-forenses: la primera elimina la posibilidad de identificar la fuente de la imagen y la segunda permite falsificar la identidad de una imagen haciendo uso de la Transformada Wavelet y del ruido del sensor. Estos dos algoritmos para la destrucción y la falsificación de la identidad de imágenes digitales se detallan en el Capítulo 13 [RCAGSO+13] [RCAGSOGV14] [RCEKSOGV15] [GVSORCHC].

Para el desarrollo de las contribuciones presentadas se ha considerado siempre una orientación práctica, habiéndose realizado además los experimentos con una amplia variedad de escenarios y dispositivos móviles.

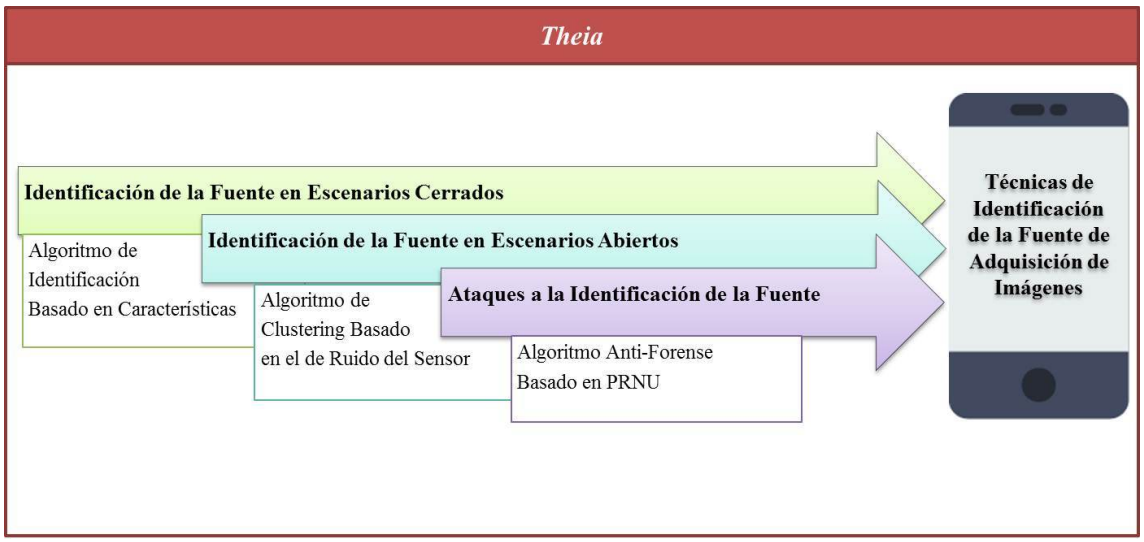


Figura 8.2: Contribuciones de la tesis

8.5 Estructura del Trabajo

Esta tesis se estructura como sigue:

El Capítulo 9 muestra algunos aspectos básicos de imágenes digitales de dispositivos móviles, presentando el proceso de adquisición y creación de imágenes en diferentes tipos

de dispositivos.

El Capítulo 10 realiza una descripción de las principales características que hacen a los dispositivos móviles fuentes potenciales de análisis forense. Se exponen asimismo las distintas ramas del análisis forense centrándose en las imágenes digitales. A continuación, se describen las principales técnicas del análisis forense mostrando los trabajos más relevantes sobre los distintos tipos de técnicas forenses de imágenes digitales [SOAGRC+13]. Seguidamente, se realiza un estudio de diversos ataques al análisis forense de imágenes digitales. Finalmente, se presenta un cuadro comparativo que resume los principales trabajos relacionados, destacando los temas de especial relevancia.

El Capítulo 11 presenta un algoritmo basado en las características del color y en las métricas de calidad de la imagen para la identificación de la fuente de adquisición de imágenes digitales [SOAGRC+13] [RCSOGV15a] [SORCGVHC16] [RCAVSOGV16]. Inicialmente se presentan los conceptos generales para la comprensión del algoritmo. Asimismo, se señalan las características y configuración de las máquinas SVM utilizadas para la clasificación. El capítulo finaliza con la presentación de los distintos experimentos realizados sobre bancos de imágenes de dispositivos móviles. Los experimentos realizados en este capítulo se han dividido en dos grupos claramente diferenciados: identificación de la fuente de adquisición (marca y modelo) e identificación del tipo de dispositivo fuente (escáner, dispositivo móvil o computador).

El Capítulo 12 desarrolla un algoritmo de agrupamiento de imágenes digitales de dispositivos móviles. Se presentan conceptos previos que facilitan la comprensión del algoritmo. Seguidamente se especifica con detalle el algoritmo propuesto y se muestran los resultados de los distintos experimentos realizados [RCAGSO+13] [RCAGSOGV14] [GVSORC15] [RCSOGV15b] [SOGVAG+15].

El Capítulo 13 expone un par de algoritmos basados en el ruido del sensor, el primero para la destrucción de la identidad de imágenes digitales y el segundo para la falsificación de la misma. El estudio de estos algoritmos permite evitar ser víctima de posibles ataques, así como fortalecer las técnicas de identificación [RCAGSO+13] [RCAGSOGV14] [RCEKSOGV15] [GVSORCHC].

El Capítulo 14 expone las principales conclusiones de este trabajo, así como algunas posibles líneas futuras de investigación.

8.6 Audiencia del Trabajo

Los requisitos previos para acceder al material de esta tesis no son elevados. Se repiten varias definiciones y conceptos disponibles en la literatura con la finalidad de hacer este trabajo independiente. Sin embargo, el lector debe tener presente que es importante tener conocimientos básicos teóricos y prácticos sobre análisis forense de imágenes digitales, cálculos estadísticos, transformada wavelet y algoritmos computacionales.

Algunos de los problemas involucrados se discuten aquí, mientras que de otros se considera que el lector ya tiene fundamentos para entender los conceptos discutidos. La bibliografía proporciona al lector información adicional para encontrar más detalles del estudio realizado en esta tesis.

Capítulo 9

Análisis Forense de Imágenes Digitales

El objetivo general de este capítulo es facilitar la compresión de las técnicas forenses que se describirán en los siguientes capítulos. Se muestran los conceptos básicos sobre el proceso de adquisición y creación de imágenes digitales en diferentes tipos de dispositivos, así como los componentes que participan en este proceso. En primer lugar se presenta el proceso de adquisición de imágenes en cámaras digitales, haciendo especial referencia a la matriz CFA y a los distintos tipos de sensores. Posteriormente, se expone el proceso de adquisición de imágenes en escáneres. El capítulo finaliza con una breve síntesis de lo expuesto en el mismo.

9.1 Introducción

Para introducirse al análisis forense de imágenes digitales el primer paso es conocer cómo se componen los dispositivos que generan la imagen y cuál es el proceso de creación de las imágenes digitales, también conocido como *pipeline*. Generalmente el *pipeline* presenta notables diferencias entre los distintos tipos de dispositivos. Dentro del mismo tipo de dispositivo la estructura del *pipeline* es semejante y varía en algunos aspectos debido al fabricante, la calidad de los componentes de la cámara o las funcionalidades que éstas ofrecen.

A continuación se muestra la estructura general del proceso de generación de una imagen en cámaras digitales y escáneres. Dentro del apartado de cámaras digitales se hace hincapié en los aspectos relevantes relacionados con las cámaras digitales de dispositivos móviles.

9.2 Proceso de Adquisición en Cámaras Digitales

En general, las cámaras fotográficas digitales se componen de un sistema de lentes, un grupo de filtros, una matriz de filtro de colores o CFA, un sensor de imagen y un procesador de imagen o DIP [BSM08]. Muchos de los detalles del proceso de generación de una imagen en una cámara digital pertenecen a cada fabricante y a cada tipo de dispositivo

considerándose información confidencial; sin embargo, existe una estructura general del mismo para cada tipo de dispositivo. La estructura general del *pipeline* de una cámara digital se resume en la Figura 9.1.

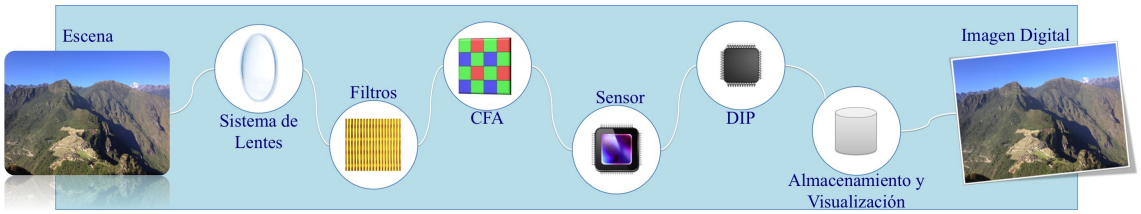


Figura 9.1: Proceso de adquisición de imágenes en cámaras digitales

Para generar una imagen en primer lugar un sistema de lentes capta la luz de la escena controlando la exposición, el foco y la estabilización de la misma. Después, la luz pasa por un grupo de filtros que mejoran su calidad. Este grupo incluye al menos un filtro infrarrojo y un filtro anti efecto de diente de sierra. El filtro infrarrojo absorbe o refleja la luz permitiendo que sólo la parte visible del espectro pase a la siguiente fase, evitando que la radiación infrarroja ocasione pérdida de nitidez en la imagen. El filtro anti efecto de diente de sierra se encarga de limpiar la señal produciendo imágenes con contornos más suaves.

A continuación la luz pasa al sensor de imagen. Puede haber mecanismos que interactúen con el sensor para determinar la exposición (tamaño de apertura, velocidad de obturación, control de ganancia automático) y la distancia focal de la lente. El sensor de imagen es una matriz de elementos sensibles a la luz llamados píxeles, los cuales son monocromáticos. Cada elemento de esta matriz de píxeles toma la luz incidente y genera una señal analógica proporcional a la intensidad de la luz recibida. Esta señal analógica se convierte a una señal digital y se transmite al procesador de imagen.

Debido a que el sensor de la imagen es monocromático, para capturar una imagen a color se requieren diferentes sensores. Idealmente, un sensor para cada color. Sin embargo, debido al coste que esto implica, en la práctica las cámaras normalmente usan un único sensor de imagen junto a una matriz CFA que se coloca antes del sensor para producir los colores.

Una vez que el procesador de imagen recibe la señal digital generada por el sensor, se elimina el ruido y otras anomalías introducidas en las señales digitales (*artifacts*), con la finalidad de obtener una imagen visualmente más agradable. El más destacado de estos procesos es el denominado interpolación cromática (*demosaicing* o *demosaicking*). El algoritmo de interpolación cromática se encarga de calcular los valores de los colores faltantes debido a que el sensor únicamente proporciona información sobre una cierta cantidad de colores (los que permite pasar la matriz CFA). Éste es uno de los procesos más complejos desde el punto de vista computacional y las técnicas utilizadas suelen ser propiedad del fabricante de la cámara.

Posteriormente, se realizan procesos como la corrección de píxeles defectuosos, el balanceo de blancos y la corrección gamma. La corrección de píxeles defectuosos originados por imperfecciones en el sensor corrige estos píxeles mediante interpolación. El balanceo de blancos permite una reproducción más fiel del color, sin que haya colores dominantes que son especialmente notables en tonos neutros como el blanco. La corrección gamma ajusta los valores de intensidad de la imagen. Aunque los algoritmos para llevar a cabo estos procesos están presentes en todas las cámaras, los detalles exactos de la forma de realizarlos pueden variar entre los diferentes fabricantes, e incluso, entre los distintos modelos de un mismo fabricante.

Finalmente, la imagen generada por el procesador se comprime. En las cámaras de dispositivos móviles normalmente se utiliza el algoritmo JPEG [Ham92] para ahorrar espacio, almacenándose en la memoria permanente del dispositivo junto con los metadatos de la imagen en formato Exif [RSYD05].

9.3 Proceso de Adquisición en Escáneres

Los escáneres constan de un *pipeline* con notables diferencias con respecto al de las cámaras fotográficas. En la Figura 9.2 se muestra la estructura general del *pipeline* de un escáner.

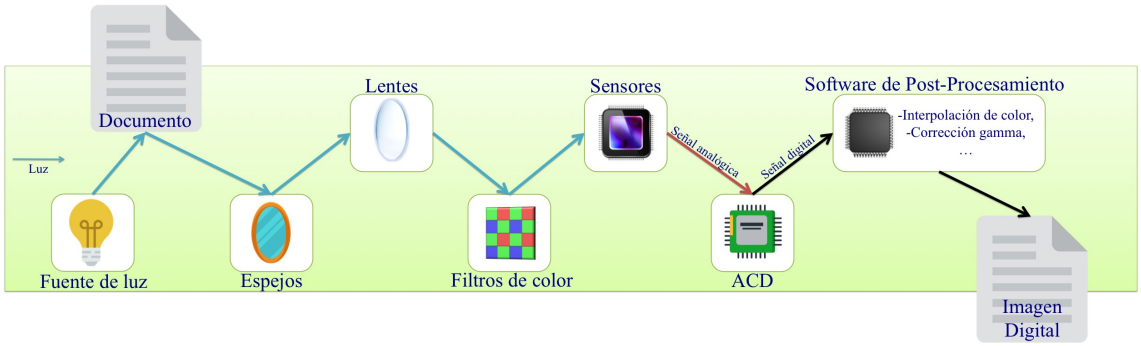


Figura 9.2: Proceso de adquisición de imágenes en escáneres

Inicialmente se utiliza una lámpara para iluminar el documento que se quiere digitalizar. Ésta puede ser una CCFL, una lámpara de xenón o en los antiguos modelos de escáneres lámparas fluorescentes normales. Existe una barra de estabilización que, junto con un motor eléctrico, consiguen mover la cabeza lectora linealmente sin tener desviaciones de ningún tipo. La máxima resolución de un escáner es definida por el número de elementos en el sensor lineal (resolución horizontal) y el tamaño de paso del motor que mueve la cabeza lectora (resolución vertical). La luz reflejada por el documento al incidir la luz de la lámpara se transmite mediante un conjunto de espejos hasta llegar a las lentes.

Las lentes son las encargadas de enfocar la luz a un conjunto de filtros de color. Cada filtro de color deja pasar la luz de un solo color a una línea de sensores, ya que éstos, al igual que en las cámaras digitales, son monocromáticos. Cada elemento de esta línea de sensores generalmente está formada por tres sensores individuales cada uno con su correspondiente filtro de color. Cada sensor, por tanto, capta cada uno de los colores del modelo aditivo (rojo, verde, azul). La mayoría de escáneres de escritorio utilizan sensores CCD, aunque también existen escáneres con sensores CMOS, CIS o PMT. De los sensores se obtiene una señal analógica que se transforma en digital en el módulo ADC.

Una vez que se tiene digitalizada la imagen, posteriormente se realiza el proceso de interpolación de la misma. La idea básica del proceso de interpolación es análogo al de las cámaras digitales, pero con las peculiaridades de los filtros y sensores del escáner. Es decir, por procesamiento software para cada píxel se genera su color a partir de los tres colores básicos captados por cada sensor y en ciertos casos con los colores de los píxeles adyacentes. Cada fabricante normalmente tiene su algoritmo de interpolación. En ciertos casos este proceso de interpolación en los escáneres se utiliza además para incrementar la percepción de la resolución de la imagen, es decir, crear píxeles nuevos no captados por el sensor mediante procesamiento software, lo cual es conocido como resolución interpolada. Finalmente, como últimos pasos antes de la creación de la imagen digital final, se producen procesamientos software para la corrección de posibles defectos de la imagen como el balanceo de blancos y la corrección gamma.

9.4 Estructura Global de la Imagen

9.4.1 Matriz de Filtros de Color

La matriz CFA es un mosaico de diminutos filtros de color colocados sobre los píxeles de los sensores de imagen para capturar la información del color y es una de las partes más importantes en el *pipeline* de una cámara [APS98]. La matriz CFA se encuentra sobre el sensor monocromo y su función es adquirir la información del color de la escena. Cada celda del filtro de color deja pasar la luz de acuerdo a un rango de longitudes de onda, de tal manera que las intensidades filtradas separadas incluyen información sobre el color de la luz. Como se ilustra en un ejemplo en la Figura 9.3, la intensidad de la luz que pasa por cada una de las celdas forma una imagen en escala de grises y, dependiendo de la configuración del filtro CFA, se interpreta como una imagen a color (considerando que cada píxel corresponde a un valor de intensidad).

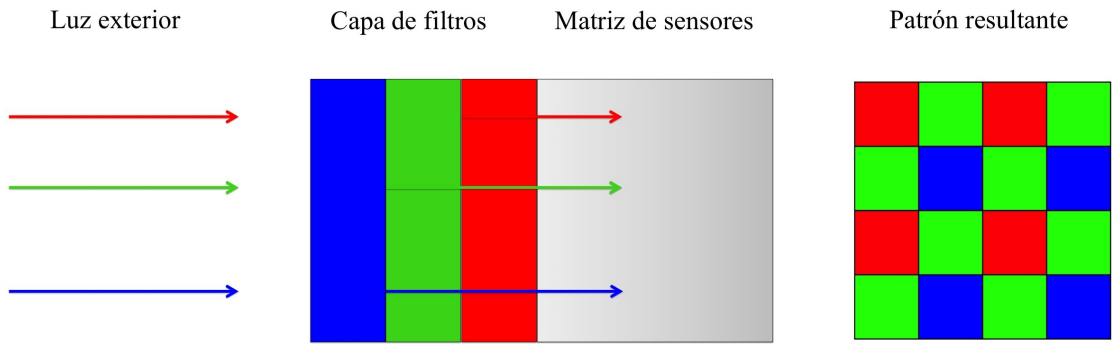


Figura 9.3: Matriz CFA

Existen valores que no se miden para cada uno de los colores del filtro CFA de acuerdo a su configuración. El algoritmo de interpolación cromática es el encargado de calcular esos valores faltantes mediante la interpolación de los valores de los píxeles vecinos. Este es el proceso más complejo en cuanto a cómputo se refiere.

El diseño de los filtros de color y el algoritmo de interpolación cromática pueden variar entre fabricantes, éste último incluso cuando se utiliza el mismo tipo de matriz CFA. El diseño de la matriz CFA influye en la imagen resultante, tanto en la nitidez como en la apariencia de los bordes y en los pequeños detalles. El proceso de interpolación cromática puede generar anomalías en la imagen tales como efecto de diente de sierra en los contornos de las imágenes, ruido y distorsiones en el color. El uso de otro filtro puede eliminar la presencia de estas imperfecciones en determinadas áreas de la imagen a costa de degradar la calidad en otras [LP05].

Existen distintos patrones CFA como el modelo GRGB de Bayer mostrado en la Figura 9.3. Otros modelos de patrón CFA son el filtro RGBE, CYYM, CYGM o el RGBW. En la Figura 9.4 se muestran los filtros de color citados anteriormente [Nak05].

Generalmente, las cámaras usan el modelo GRGB del patrón de Bayer y la generación de la imagen final por parte del algoritmo de interpolación cromática. Este filtro captura para el canal rojo el 25 % de los píxeles, para el verde el 50 % y para el azul el 25 % restante. Esto significa que para la construcción de la imagen final se tiene que recuperar el 75 % de los píxeles del canal rojo, el 50 % del canal verde y el 75 % del canal azul. En la Figura 9.5 puede verse un ejemplo de captura de color con este esquema.

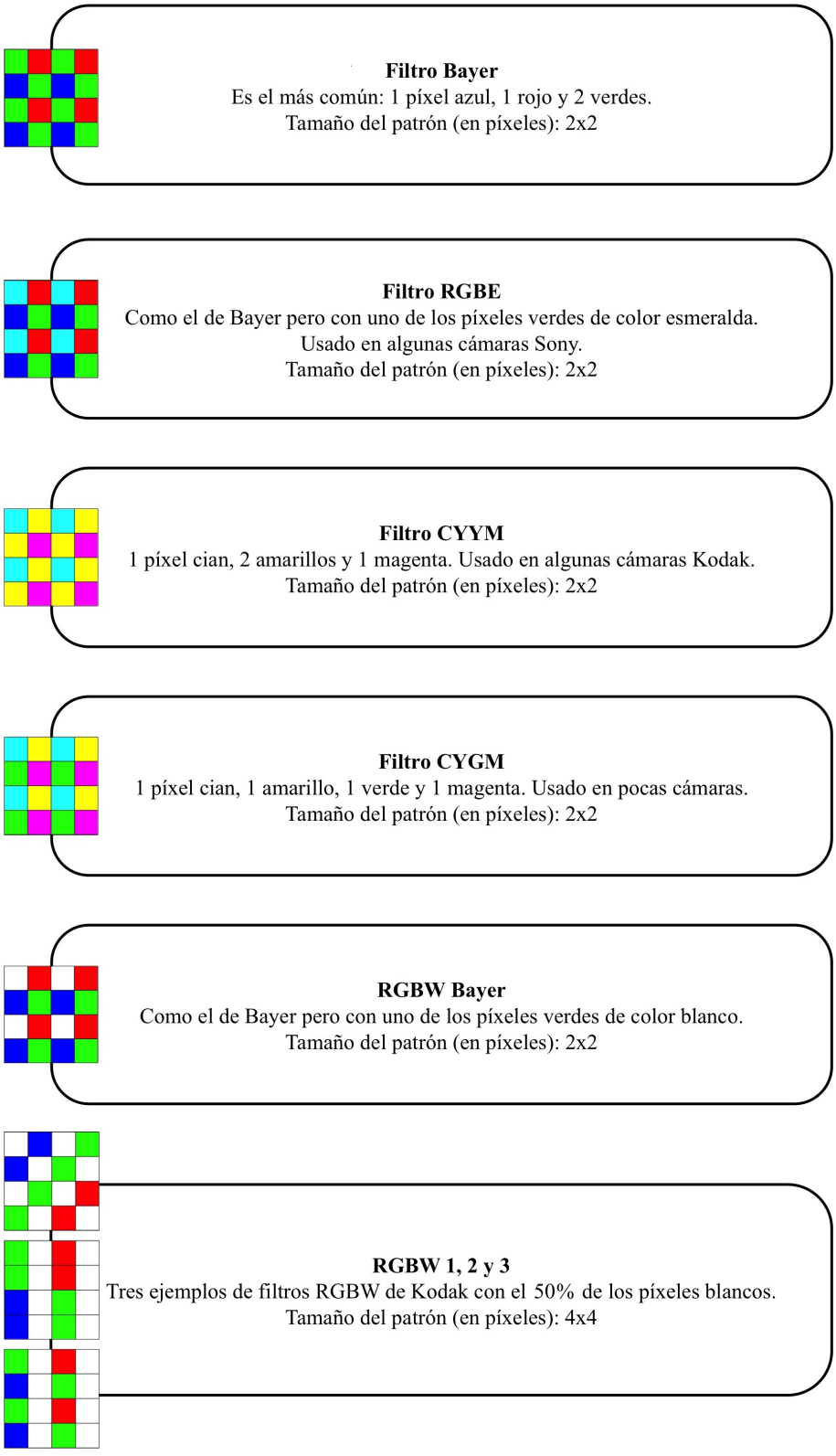


Figura 9.4: Tipos de filtros de color

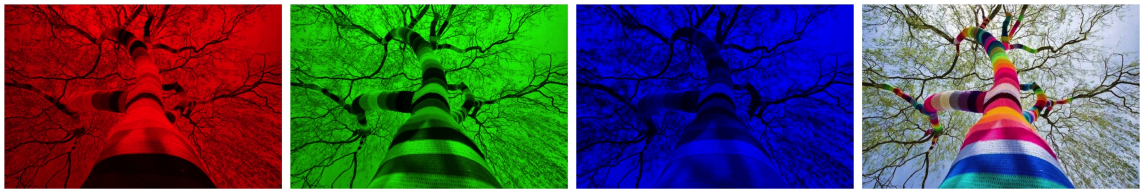


Figura 9.5: Ejemplo de aplicación del filtro GRGB de Bayer a una imagen real

9.4.2 Sensores de Imagen

El sensor de imagen es la parte más importante de las cámaras digitales. Generalmente, es considerado el corazón de la cámara. El sensor es una matriz de elementos sensibles a la luz llamados píxeles. Los píxeles están hechos de silicio y capturan la luz convirtiendo los fotones en electrones utilizando el efecto fotoeléctrico. Cada píxel se encarga de acumular la carga inducida por la luz durante un determinado tiempo de exposición para luego ser leído y procesado. La señal de salida del sensor es proporcional a la carga acumulada, dependiendo de la cantidad de luz que incida sobre el píxel y del tiempo de exposición a ella.

En la actualidad existen dos tipos de tecnologías utilizadas para la fabricación de sensores de cámaras digitales: CCD y CMOS. Ambos tipos de sensores están formados esencialmente por semiconductores de metal-óxido (MOS), distribuidos en forma de matriz y funcionando de forma similar. Sin embargo, hay características que diferencian a estas tecnologías. La diferencia clave entre las dos tecnologías de sensores es el lugar en el que se digitalizan los píxeles y la forma en la que se lleva a cabo la lectura de las cargas.

9.4.2.1 Sensores CCD

En los sensores CCD cada una de las cargas de las celdas de la matriz se transforman en voltajes y se entrega una señal analógica como salida para que posteriormente se digitalice por la cámara. La estructura de este tipo de sensores es muy sencilla, pero tiene como inconveniente la necesidad de contar con un chip adicional que trate la información de salida del sensor (implicando equipos más grandes y costosos).

A diferencia de los sensores CMOS que soportan la lectura de la matriz de píxeles de una manera aleatoria, en los sensores CCD todos los píxeles comienzan y finalizan la integración de carga al mismo tiempo. Esto propicia una salida uniforme (resultado que se espera de un píxel sometido al mismo nivel de excitación de los demás sin que se presenten cambios notables en la señal obtenida). A este tipo de exposición se le conoce como lectura de obturador global. Es posible añadir circuitos en los sensores CMOS para hacer que den un resultado similar. Sin embargo, los sensores CMOS siguen siendo superiores a los sensores de tipo CCD.

Los sensores CCD son, por mucho, mejores que los de tipo CMOS en cuanto al rango dinámico (coeficiente entre la saturación de los píxeles y el umbral por debajo del cual no captan señal), puesto que al ser menos sensibles toleran mejor los extremos de luz.

Asimismo, los sensores CCD son superiores a los CMOS en términos de ruido en la imagen, puesto que el procesado de las señales se lleva a cabo en un chip externo que puede optimizarse para el desarrollo de esta función. Por el contrario, los sensores CMOS realizan el procesamiento de la señal dentro del mismo sensor dejando menos espacio para colocar los foto-diodos encargados de recolectar la luz.

9.4.2.2 Sensores CMOS

Los sensores CMOS son sensores con un diseño de píxeles activos e independientes. Se denominan píxeles activos debido a que la digitalización se realiza en ellos internamente en unos transistores que ofrecen mejor velocidad de procesamiento, eliminándose la necesidad de un chip externo que realice esta función, lo que reduce el coste y el tamaño de los equipos.

La característica de independencia se refiere a la flexibilidad que este tipo de sensores ofrece para la lectura de la matriz de píxeles, ya que es posible acceder a cada celda mediante la posición de su fila y columna. Generalmente, la lectura de la matriz se realiza con el esquema conocido como lectura de obturador en barrido progresivo (no es necesario leer la matriz completa en un solo tiempo como en los sensores CCD). Además, al estar formados por celdas independientes, los sensores CMOS no presentan el efecto *blooming*. Este efecto se produce cuando un píxel se satura por la luz que incide sobre él y a continuación comienza a saturar a los que se encuentran a su alrededor.

Una ventaja más es que los sensores CMOS son más sensibles a la luz y en condiciones de poca iluminación se comportan mejor. Adicionalmente, debido a que los amplificadores de la señal se encuentran dentro de la misma celda, no se genera un consumo extra de alimentación a diferencia de los sensores CCD.

En sus inicios los sensores CMOS no eran considerados tan buenos como los sensores CCD. Sin embargo, la tecnología CCD ha llegado a su límite y ahora es cuando se está desarrollando la tecnología CMOS, superándose día a día sus deficiencias, tanto es así que existen los denominados sensores *Smart* CMOS [Oht07] con el objetivo de mejorar los defectos de los sensores CCD y CMOS convencionales.

La mayoría de las DSCs utilizan los sensores CCD. En dispositivos móviles es más común el uso de los sensores CMOS. Incluso día a día, reduciendo las diferencias de calidad entre los sensores CCD y CMOS, en la gran mayoría de casos los sensores DSCs sobrepasan notablemente en calidad a los sensores de las cámaras digitales de dispositivos móviles, y esta es la principal razón para requerir técnicas específicas para la fuente de adquisición de la imagen en dispositivos móviles. Del mismo modo que en el caso de los sensores, las lentes de las cámaras digitales de dispositivos móviles son, en general, de menor calidad que las lentes de las DSCs.

9.4.2.3 Imperfecciones y Ruido del Sensor

Las imperfecciones son anomalías que permanecen constantes de una imagen a otra, mientras que el ruido se considera una anomalía aleatoria, al igual que la estática en un televisor. El ruido del sensor no sobrevivirá al promediado de fotogramas, mientras

que las imperfecciones del sensor lo harán. Sin embargo, generalmente en los algoritmos descritos a continuación, a veces las imperfecciones del sensor se denominan ruido del sensor, señalando a las características que permanecen constantes.

La generación de imágenes para cualquier dispositivo es compleja y varía dependiendo del equipo y del fabricante. Sin embargo, hay tipos similares de ruidos que son inherentes a cada dispositivo, tanto aleatorios como sistemáticos. Los ruidos de disparo y de cuantificación son erráticos y no tienen patrones consistentes o predecibles. El ruido de disparo es el resultado del flujo no continuo de corriente eléctrica: la suma de impulsos discretos en el tiempo para cada píxel.

Cuanto más tiempo esté activo el sensor, es decir, mientras mayor sea la velocidad del obturador o mientras más sensible sea el sensor (como en condiciones de luz baja), mayor ruido de electrones al azar se grabará junto con la escena por el sensor de imagen. Este tipo de ruido depende de la temperatura (altas temperaturas causarán un mayor movimiento de los electrones en los circuitos). El ruido de cuantificación es causado por el proceso de convertir la luz de una cantidad infinita de valores de intensidad en un medio digital que tiene una cantidad finita de niveles de intensidad. Si bien este proceso introduce pequeñas distorsiones en la imagen, un detalle más fino con mayor profundidad de bits puede minimizar este error.

La mayoría de las investigaciones sobre el análisis forense de la fuente de adquisición de imágenes se centran en las cámaras digitales tradicionales o DSCs. La mayoría de estas técnicas no son válidas para las imágenes de dispositivos móviles. La razón principal es que la mayoría de las técnicas se basan en el uso directo o indirecto de las características del sensor o en la lente de la cámara digital. Respecto al sensor, es el componente que se encarga de capturar la luz y generar una señal digital según su intensidad.

Los componentes principales del ruido de la imagen son el FPN y el PRNU. Hay varias fuentes de imperfecciones y del ruido introducido en las diferentes etapas del *pipeline* de la creación de una imagen en una cámara digital. Incluso si se toma una imagen uniforme y completamente iluminada, es posible ver pequeños cambios en la intensidad entre píxeles. Esto se debe a que el ruido de disparo es aleatorio y, en gran parte, el ruido del patrón es determinista y se mantiene aproximadamente igual si se toman varias imágenes de la misma escena.

El patrón de ruido de una imagen se refiere a cualquier patrón espacial que no cambia de una imagen a otra. Se compone del ruido que es independiente de la señal (FPN) y del ruido debido a la diferencia de respuesta de cada píxel a la señal incidente (PRNU). La estructura del patrón de ruido se muestra en la Figura 9.6.

El ruido FPN se genera por la corriente de oscuridad y también depende de la exposición y de la temperatura. Debido a que el FPN es un ruido independiente aditivo, algunas cámaras lo eliminan automáticamente restando un marco oscuro a las imágenes que generan.

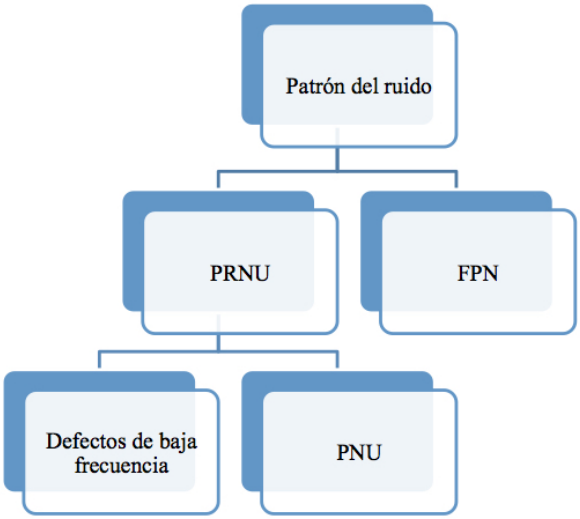


Figura 9.6: Patrón del ruido del sensor

9.4.2.4 No Uniformidad en la Respuesta Fotónica

El ruido PRNU es la parte dominante del patrón de ruido de las imágenes y es un ruido dependiente multiplicativo. El ruido PRNU está formado principalmente por la uniformidad de píxel PNU y los defectos de baja frecuencia como la configuración del *zoom* y la refracción de la luz en las partículas de polvo y en las lentes. El ruido PNU es la diferencia de sensibilidad a la luz entre los píxeles de la matriz del sensor. Se genera por la falta de homogeneidad de las obleas de silicio y las imperfecciones durante el proceso de fabricación del sensor. Debido a su naturaleza y origen es muy poco probable que incluso los sensores procedentes de la misma oblea presenten patrones PNU correlacionados. Este ruido no se ve afectado por la temperatura ambiente ni por la humedad. El ruido PNU es normalmente más común, complejo y significativo en los sensores CMOS debido a la complejidad de la circuitería de la matriz de píxeles.

9.5 Síntesis del Capítulo

El objetivo de este capítulo ha sido introducir el proceso de adquisición de imágenes digitales. El capítulo ha comenzado con la explicación del *pipeline* para la generación de imágenes en cámaras digitales. Se ha hecho especial hincapié en los elementos del *pipeline* que podrían estudiarse para el desarrollo de futuras técnicas forenses de identificación de la fuente de adquisición. Se ha realizado una explicación sobre la matriz CFA, ya que se considera que la especificación de ésta junto con el algoritmo de interpolación de color utilizado producen las diferencias más significativas entre los diferentes modelos de cámara. De la misma manera, se ha llevado a cabo una descripción de los diferentes tipos de sensores, con el fin de aclarar las diferencias en este elemento entre DSCs y cámaras digitales en dispositivos móviles. Finalmente, se ha mostrado el *pipeline* de escáneres, que, a pesar de tener elementos comunes al de una cámara digital, difiere notablemente.

Capítulo 10

Técnicas de Análisis Forense en Imágenes Digitales

Este capítulo presenta el estado del arte clasificando el trabajo relacionado en los siguientes grupos: técnicas para identificar la fuente de adquisición de imágenes, técnicas de agrupamiento de imágenes y ataques al análisis forense de imágenes digitales. Comienza con la clasificación del análisis forense y continúa con la exposición del trabajo relacionado con la identificación de la fuente de adquisición de imágenes digitales. Posteriormente, se presenta el trabajo relacionado con el agrupamiento de imágenes. Después, una sección muestra la clasificación de las contramedidas forenses y las investigaciones relacionadas con los ataques al análisis forense de imágenes digitales. Cabe señalar que aunque este trabajo se centra en dispositivos móviles, en el estado del arte se incluyen referencias a técnicas sobre imágenes de todo tipo de dispositivos, ya que el conocimiento de éstas puede ser válido para la aplicación o adaptación a imágenes generadas por dispositivos móviles. El capítulo finaliza con una breve síntesis de lo expuesto en el mismo.

10.1 Técnicas de Análisis Forense en Imágenes

En esta sección se describen las principales técnicas de análisis forense de imágenes digitales para la identificación de la fuente de adquisición de la imagen y los principales trabajos del estado del arte. Otros compendios de técnicas pueden verse en [SWL09], [RSBG11] y [SOAGRC+13].

De acuerdo a [CFGL08] las tareas de análisis forense de imágenes digitales se pueden dividir en las siguientes categorías:

- **Clasificación basada en la fuente**: Tiene como objetivo clasificar las imágenes de acuerdo a su origen en cámaras digitales o escáneres.
- **Identificación de la fuente**: Busca determinar el dispositivo específico que generó una imagen determinada (marca y modelo).
- **Agrupamiento por dispositivo fuente**: Dado un grupo de imágenes se buscan los grupos de imágenes que fueron obtenidas utilizando la misma cámara.

- **Recuperación de la historia de procesamiento**: Tiene como objetivo recuperar la cadena de procesamientos que han sido aplicados a una imagen de una manera no maliciosa como, por ejemplo, recortes, filtrados, contrastes, etc.
- **Verificación de integridad o detección de falsificaciones**: Busca descubrir procedimientos maliciosos que se hayan aplicado a las imágenes como, por ejemplo, recorte o adición de objetos a una imagen.

Esta clasificación se resume en la Figura 10.1.

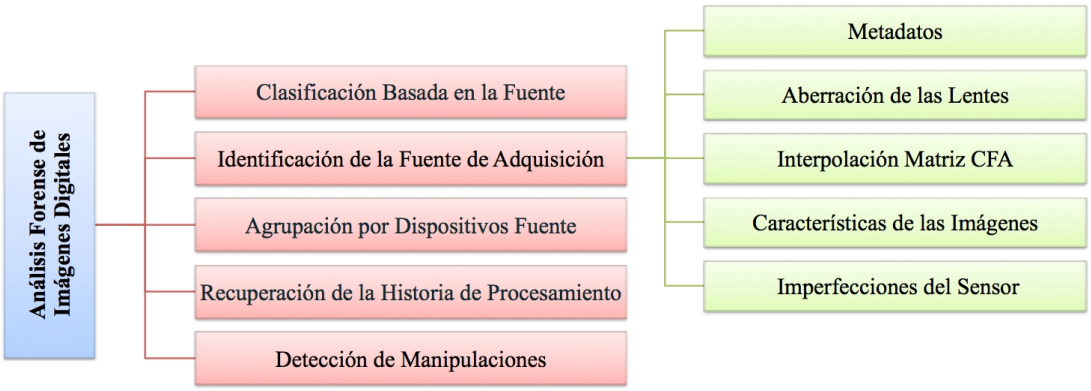


Figura 10.1: Clasificación de las técnicas de análisis forense en imágenes

El éxito de estas técnicas depende del supuesto de que todas las imágenes adquiridas por un mismo dispositivo presentan características intrínsecas del mismo. Las características que se usan para identificar la marca y el modelo de las cámaras digitales se derivan de las diferencias que existen entre las técnicas de procesamiento de las imágenes y las tecnologías de los componentes que se utilizan. El mayor problema con este enfoque es que los diferentes modelos de las cámaras digitales usan componentes de un número reducido de fabricantes, y que los algoritmos que usan también son muy similares entre modelos de la misma marca. Según [VLCEK07] se pueden establecer cuatro grupos de técnicas para este fin: basadas en la aberración de las lentes, basadas en la interpolación de la matriz CFA, basadas en el uso de las características de la imagen y basadas en las imperfecciones del sensor. Además de los anteriores, existe otro grupo de técnicas forenses a destacar basadas en los metadatos de la imagen.

10.1.1 Técnicas Basadas en Metadatos

Las cámaras digitales cuentan con una poderosa fuente de información que son los metadatos embebidos en los archivos de las imágenes digitales. Los metadatos o "datos sobre datos" registran información relacionada con las condiciones de captura de la imagen, como fecha y hora de generación, presencia o ausencia de *flash*, distancia de los objetos, tiempo de exposición, apertura del obturador e información GPS, etc. En otras palabras, información de interés que complementa el contenido principal de un documento digital.

La especificación Exif [Sta10] es el contenedor de metadatos más común en las cámaras digitales [Bae10]. La especificación Exif incluye cientos de etiquetas, entre las que se encuentran *marca* y *modelo*, aunque cabe destacar que la propia especificación no hace obligatoria su existencia en los archivos de las imágenes.

Las técnicas basadas en los metadatos de la imagen son las más sencillas. Existen gran cantidad de trabajos enfocados sobre los diferentes tipos de metadatos, tanto para la búsqueda de información, como para la clasificación de imágenes [Pla00] [BL05] [Tes05] [RCC+08] [AG11]. Asimismo, los metadatos pueden utilizarse como datos de entrada o ayuda para el uso de otras técnicas forenses. Por ejemplo, en la aplicación de técnicas basadas en el contenido de la imagen, los metadatos Exif puede ofrecer una gran cantidad y variedad de información de aspectos técnicos, que pueden permitir aumentar las tasas de acierto o mejorar los resultados de la aplicación de ciertos algoritmos forenses [BL04] [JLLC07] [FKCS11].

Sin embargo, estas técnicas dependen en gran medida de los metadatos que los fabricantes deciden insertar cuando la imagen es generada y en la corrección de los mismos. En [SOAGGVHC12] y [SOAGGVHC14] se realiza un estudio a fondo sobre este tema. Asimismo, este método es el más vulnerable a modificaciones malintencionadas.

10.1.2 Técnicas Basadas en la Aberración de las Lentes

Durante el proceso de generación de la imagen en la parte del sistema de lentes se pueden introducir aberraciones. Existen diferentes tipos de aberraciones: aberración esférica, coma o aberración comática, astigmatismo, curvatura de campo, distorsión radial y distorsión cromática. La distorsión radial es la que más consecuencias tiene sobre la imagen, especialmente en las cámaras que usan lentes baratas de gran angular. La mayoría de cámaras digitales usan este tipo de lentes por cuestiones de coste.

En [Cho06] se propone la distorsión radial de la lente como técnica para la identificación de la fuente. Los autores concluyen que cada modelo de cámara expresa un único patrón de distorsión radial que ayuda a identificarla de manera única. En los experimentos se utilizaron tres cámaras diferentes y obtuvieron como resultado una precisión del 87 % al 91 % en la identificación de la fuente.

En [VEK07] se propone la aberración cromática lateral como técnica para la identificación de la fuente. Los autores realizan distintos tipos de experimentos utilizando conjuntos no numerosos de cámaras con imágenes no modificadas, modificadas o con recortes aleatorios en regiones de la imagen. En el experimento en el que se usaron tres cámaras de diferentes marcas se obtuvo una precisión del 86,67 % en la identificación de la fuente. Finalmente, se concluye que esta técnica no es adecuada para la identificación de la fuente de distintos modelos de cámara de la misma marca.

10.1.3 Técnicas Basadas en la Interpolación de la Matriz CFA

Algunos autores consideran que la elección de la matriz de colores CFA y la especificación de los algoritmos de interpolación cromática generan algunas de las diferencias más marcadas entre los diferentes modelos de cámaras [BSM06] [CAS+06]

[LH06] [BSM08].

Las cámaras comerciales tienen un solo sensor en lugar de varios sensores para cada componente del color. En esencia, la interpolación cromática introduce un tipo específico de correlación entre los valores de colores de los píxeles de la imagen. La forma específica de estas dependencias se pueden extraer de las imágenes para diferenciar los algoritmos de interpolación cromática y así determinar marca y modelo de la cámara que generó una imagen.

En [LH06] se utilizan las correlaciones entre píxeles en el proceso de identificación de la fuente. Para la clasificación utilizan redes neuronales. Se prueba el método para cuatro cámaras y la tasa de acierto oscila entre el 95 % y el 100 %, con una tasa media del 98,25 %. También se realizan pruebas para imágenes modificadas, con resultados de un 80 % de éxito para una compresión JPEG del 80 %. Dado que las cámaras del mismo fabricante utilizan el mismo algoritmo de interpolación cromática, esta técnica no es eficiente para diferenciar entre distintos modelos del mismo fabricante. Asimismo, no se obtienen buenos resultados cuando las imágenes han sido modificadas o tienen un nivel alto de compresión.

En [CAS+06] se utiliza un conjunto de medidas de similitud binarias como métricas para estimar la semejanza entre los planos de bits de una imagen. En este estudio se utilizan 108 medidas de similitud binarias. Los experimentos realizados con esta técnica para clasificar 3 grupos de cámaras obtienen un porcentaje de éxito entre el 81 % y el 98 %, mientras que para un grupo de 9 cámaras la precisión desciende al 62 %. Claramente se puede apreciar que los resultados del método dependen del número de cámaras utilizadas en los experimentos.

En [BSM08] se presenta un algoritmo para identificar y clasificar las operaciones de interpolación cromática. La propuesta se basa en dos métodos para realizar el proceso de clasificación: el primer método utiliza un algoritmo para analizar la correlación del valor de cada píxel con los valores de sus vecinos y el segundo método realiza un análisis de las diferencias entre píxeles. Se realizan experimentos con diferentes números de cámaras y distintos tipos de imágenes. Los resultados obtenidos en la identificación de la fuente de una imagen varían entre el 84,8 % y el 92,56 % de tasa de acierto medio.

En [CK10] se presenta una técnica para la identificación de la fuente basada en la información del proceso de interpolación de la matriz CFA y una comparativa con otras técnicas. Esta técnica presenta tres nuevos grupos de características de interpolación cromática: pesos, EC y NGS. Dado que el número de características es muy alto se realiza un proceso ERE para disminuir el número de las mismas. Se realizan distintos experimentos utilizando clasificadores 1NN y PSVM. Los resultados utilizando 15 cámaras de 4 fabricantes distintos y 11 modelos diferentes (hay cámaras de la misma marca y mismo modelo), con una reducción a 20 características y un clasificador PSVM, obtienen unos resultados de acierto medio del 99,4 % para la distinción de la marca y un 94,8 % para la del modelo.

En [HAZG10] se proponen cuatro algoritmos que utilizan aspectos basados en la correlación entre canales. Estos algoritmos calculan mapas de varianza y los clasifican utilizando 1NN. En los experimentos para la identificación de la fuente de la imagen se utilizan cuatro cámaras de tres fabricantes distintos y 50 imágenes de cada una (25 para

entrenar y 25 para pruebas). Los resultados muestran un acierto medio del 94,5 % y los autores concluyen que la correlación entre canales ofrece un enfoque complementario a trabajos anteriores que tratan correlaciones entre los píxeles introducidas por el proceso de interpolación cromática.

10.1.4 Técnicas Basadas en las Características de las Imágenes

Estas técnicas utilizan un conjunto de características extraídas del contenido de la imagen para hacer la identificación de la fuente. Estas características se dividen en tres grupos: características de color, métricas de calidad de la imagen IQM y estadísticas del dominio wavelet.

En [TLL07] se propone un método de identificación de la fuente utilizando las siguientes características: color, calidad de la imagen y dominio de la frecuencia. En el estudio adoptan la transformada wavelet como método para calcular las estadísticas del dominio wavelet y utilizan SVM para la clasificación. En los experimentos realizados se usan cámaras digitales y dispositivos móviles. Los resultados obtenidos en los distintos experimentos arrojan unos resultados entre el 61,7 % y el 99,72 % de acierto.

En [MSGW08] se extiende la identificación de la fuente a diferentes dispositivos tales como teléfonos móviles con cámara integrada, cámaras digitales, escáneres y computadores. En esta propuesta se usan las diferencias en el proceso de adquisición de la imagen de los dispositivos para formar dos grupos de características: coeficientes de interpolación de color y características de ruido. En los experimentos se utilizan cinco modelos de teléfonos móviles, cinco modelos de cámaras digitales y cuatro modelos de escáneres para identificar el tipo de fuente. En los resultados globales se obtiene un 93,75 % de precisión. En el análisis de identificación de marca y modelo de teléfonos móviles se obtiene una precisión del 97,7 % para los cinco modelos.

En [WGKM09] se propone un método para la identificación de la cámara fuente mediante la extracción y clasificación de las estadísticas de las características wavelets. Finalmente se obtiene 216 características wavelet de primer orden y 135 características de co-ocurrencia de segundo orden. Se seleccionan las características más representativas utilizando un algoritmo SFFS y se clasifican utilizando SVM. Se consigue una media de identificación del 98 % para el conjunto de todas las cámaras y una tasa de acierto media del 96,9 % para las tres cámaras del mismo modelo.

En [HLZ10] se realizan experimentos con las características de las imágenes para la identificación de la fuente más usuales: wavelet, color, IQM, características estadísticas de la diferencia de imágenes y características estadísticas de predicción de errores. En los experimentos se proponen distintas combinaciones de los distintos tipos de características y SVM para la clasificación de los distintos dispositivos. Se utilizan diez cámaras diferentes de cuatro fabricantes distintos con 300 imágenes de cada cámara (150 para entrenamiento y 150 para probar) y de una resolución de 1024×1024. Utilizando todas las características se obtiene un resultado de acierto medio del 92 %. Asimismo, se realizan experimentos para comprobar la robustez ante tres de las manipulaciones más comunes en imágenes digitales: la compresión JPEG, el recorte y el escalado. Las conclusiones finales de este trabajo son que algunos de los conjuntos de características ofrecen buenas tasas de aciertos

para imágenes intactas, pero no para las que tienen modificaciones. También se concluye que distintos tipos de manipulaciones tienen efectos diferentes sobre las tasas de acierto de los distintos conjuntos de características.

En [OA11] se plantea una técnica para la identificación de la fuente de una imagen utilizando los modelos estadísticos para *ridgelet* y sub-bandas *contourlet*. Tras la extracción de las características se aplica un algoritmo SFFS para selección de características y SVM para la clasificación. El método basado en wavelets considera 216 características útiles sólo para la representación de una dimensión, el enfoque basado en *ridgelets* toma 48 características y la aproximación de *contourlets* contempla un total de 768 características. En los experimentos realizados con tres cámaras de distintos fabricantes se obtienen porcentajes de acierto entre el 99,5 % y el 99,8 %. Los *contourlets* y *ridgelets* no sólo son efectivos para diferenciar entre modelos de cámaras, sino también para diferenciar entre imágenes naturales o producidas por computador, o para diferenciar imágenes de escáneres de la misma marca. De cualquier manera los autores consideran que se pueden implementar mejoras experimentando con diferentes algoritmos de selección.

En [LLC+12] se propone un método que emplea la densidad marginal de los coeficientes de la Transformada Coseno Discreta (DCT) en las coordenadas de frecuencia baja y las características de densidad del vecindario en el domino DCT. Adicionalmente, se utiliza el agrupamiento jerárquico y SVM para detectar la fuente de adquisición de las imágenes. En los experimentos realizados con imágenes pertenecientes a cinco modelos de teléfonos inteligentes de cuatro fabricantes, se obtiene entre el 86,36 % y el 99,91 % de acierto, alcanzándose los mejores resultados con un núcleo SVM lineal.

10.1.5 Técnicas Basadas en el Uso de las Imperfecciones del Sensor

Estas técnicas se basan en el estudio de las huellas que los defectos del sensor pueden dejar sobre las imágenes. Estas técnicas se dividen en dos ramas: defectos de píxel y SPN. En la primera se estudian los defectos de píxel, píxeles calientes, píxeles muertos, defectos de fila o columna y los defectos de grupo. En la segunda se construye un patrón del ruido promediando los múltiples residuos de ruido obtenidos mediante algún filtro de eliminación de ruido. La presencia del patrón se determina utilizando algún método de clasificación como correlación o máquinas SVM.

En [GBK+01] se estudian los defectos de los píxeles en los sensores CCD, centrándose en la evaluación de diferentes características para examinar las imágenes e identificar la fuente: defectos del sensor CCD, formato de los archivos usados, ruido introducido en la imagen y marcas de agua introducidas por el fabricante de la cámara. Entre los defectos del sensor CCD considerados se encuentran los puntos calientes, los píxeles muertos, los defectos en grupo y los defectos de fila o columna. En los resultados se observa que cada una de las cámaras tiene un patrón de defecto diferente. Sin embargo, también se señala que el número de defectos en los píxeles para una cámara es diferente entre fotos y varía demasiado en función del contenido de la imagen. Asimismo, se revela que el número de defectos cambia con la temperatura. Por último, el estudio señala que las cámaras con CCD de alta calidad no tienen este tipo de problema. También es cierto que la mayoría de las cámaras tienen mecanismos adicionales para compensar este tipo de problemas. Al

considerar únicamente los defectos de los sensores CCD este estudio no es aplicable al análisis de imágenes generadas por dispositivos móviles.

En [LFG06] se analiza el patrón de ruido del sensor de un conjunto de cámaras, el cual funciona como una huella dactilar, permitiendo la identificación única de cada cámara. Para obtener este patrón se realiza un promedio del ruido obtenido a partir de diferentes imágenes utilizando un filtro de eliminación de ruido. Para identificar la cámara a partir de una imagen dada, se considera el patrón de referencia como una marca de agua cuya presencia en la imagen es establecida mediante un detector de correlación. El estudio se realiza con aproximadamente 320 imágenes procedentes de 9 cámaras (2 son exactamente del mismo modelo) y se obtienen buenos resultados. Cabe destacar que este porcentaje de éxito se debe a que en los experimentos los autores utilizan el mismo conjunto de imágenes para calcular el patrón de referencia y las correlaciones. También se demuestra que este método está afectado por algoritmos de procesamiento de la imagen, como la compresión JPEG y la corrección gamma. Según [VLCEK07] los resultados para fotografías recortadas no son satisfactorios. Asimismo, en esta técnica las imágenes de las que se extrae el patrón de referencia tienen que tener el mismo tamaño que las imágenes a probar.

En [CESR12] se propone un enfoque para la identificación fuente de la cámara considerando escenarios abiertos, donde, a diferencia de los escenarios cerrados, no se da por sentado contar con acceso a todas las posibles cámaras. Esta propuesta comprende tres fases: definición de las regiones de interés, determinación de las características e identificación de la cámara fuente. Las diferentes regiones de las imágenes pueden contener información distinta sobre la huella digital de la cámara fuente. Este enfoque, en contraste con otros, considera nueve áreas de interés (ROI) y no sólo la región central de la imagen. El uso de las regiones de interés facilita trabajar con imágenes de diferentes resoluciones. Para determinar las características se calcula el SPN para cada uno de los canales R, G, B e Y (luminancia), generándose un total de 36 características para representar cada imagen.

Después, las imágenes tomadas por la cámara bajo investigación son etiquetadas como clase positiva y las tomadas por las cámaras disponibles restantes como clases negativas. En la fase de entrenamiento de la SVM se calcula el hiper-plano que separa los casos positivos y negativos. Posteriormente, se tienen en cuenta las clases desconocidas del escenario abierto, moviendo el hiper-plano generado hacia las clases positivas o hacia las clases negativas. Mediante el movimiento del hiper-plano se puede variar el margen para determinar si una imagen pertenece a una clase u otra. A este proceso se le denomina modelado de límites de decisión. En los experimentos se utiliza un conjunto de 25 cámaras digitales de 9 fabricantes, 150 imágenes de cada cámara en formato JPEG con diferentes configuraciones de luz, *zoom* y *flash*. Los resultados de los experimentos muestran una precisión del 94,49 %, del 96,77 % y del 98,10 %, utilizando conjuntos abiertos con 2/25, 5/25, y 15/25 cámaras, respectivamente, definiendo un conjunto abierto x/y como el conjunto de y cámaras donde x cámaras son conocidas y utilizadas para entrenar e $x - y$ son las cámaras desconocidas, cuyas imágenes junto con las de las cámaras conocidas son utilizadas en la fase de predicción.

[dOCSE+14] es una extensión del artículo, donde además de presentar las otras técnicas y algoritmos, se realizan nuevos experimentos. En los experimentos realizados se utilizan

13.210 imágenes de 400 cámaras (sólo se tenía acceso físico a 25 cámaras, el resto son imágenes descargadas de Flickr) y los mejores resultados obtienen tasas de acierto del 96,56 %, 97,34 %, 96,80 % y del 97,18 %, utilizando conjuntos abiertos con 2/25, 5/25, 10/25 y 15/25 cámaras, respectivamente.

10.2 Técnicas de Agrupamiento de Imágenes Digitales

El objetivo del análisis de grupos o agrupamiento es clasificar una colección de objetos en clases representativas llamadas grupos, sin información a priori, de forma que los objetos pertenecientes a cada grupo guarden una mayor similitud con respecto a los objetos de otros grupos.

De acuerdo a la clasificación de algoritmos de agrupamiento propuesta en [Rok10], se encuentran los métodos jerárquicos cuyo propósito es lograr una estructura denominada dendograma (Figura 10.2), que representa el agrupamiento de los objetos de acuerdo a sus niveles de similitud. Este agrupamiento puede realizarse de forma aglomerativa o divisiva. El agrupamiento aglomerativo considera inicialmente a cada objeto como una clase independiente hasta, de forma iterativa, lograr agrupar todos los objetos en una clase única. El agrupamiento de forma divisiva se basa en la idea de partir de una sola clase hasta lograr separar todos los objetos en clases individuales.

También existen los algoritmos de particionamiento, en donde iniciando de una partición, el algoritmo se encarga de mover objetos de un grupo a otro hasta minimizar cierto criterio de error. Dentro de esta categoría el método más famoso es el *k-means*; sin embargo, la mayoría de estos métodos requieren conocer de antemano el número de grupos, por lo cual no son muy utilizados en temas de análisis forense de imágenes.

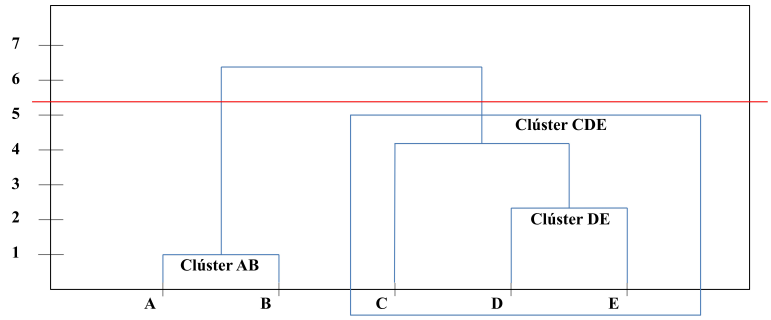


Figura 10.2: Ejemplo de dendograma

Existen otros algoritmos de agrupamiento como: [Zah71] que produce grupos por medio de grafos, [BR93] basado en la densidad donde los puntos dentro de un grupo vienen dados por cierta función de probabilidad, grupos basados en modelos como árboles de decisión [Fis87] o redes neuronales [VA00] y agrupamiento con métodos de *soft-computing* como agrupamiento difuso [HKKR99], métodos evolutivos de agrupamiento y recocido simulado en agrupamiento [SA91]. En [XW05] se muestra una amplia revisión de los distintos tipos

de algoritmos de agrupamiento, así como una extensa revisión de los enfoques utilizados sobre este tema en el estado del arte. Entre otros aspectos se concluye que no existe un algoritmo de agrupamiento universal para resolver cualquier tipo de problema y, por tanto, los enfoques de agrupamiento para cada campo o situación pueden ser totalmente diferentes. Asimismo, se destaca la importancia de la fase de selección y extracción de las características de los elementos a clasificar.

Existen trabajos previos sobre agrupamiento de imágenes por métodos sin supervisión, todos ellos consideran el ruido SPN como el criterio más fiable para representar la huella digital de un dispositivo. Más concretamente, utilizan el ruido PRNU como huella y la correlación normalizada como medida de similitud para lograr el agrupamiento de imágenes por dispositivo.

En [LLHC10] se utiliza una técnica de clasificación con aprendizaje no supervisado donde mediante la maximización de grafos se logra un agrupamiento. El agrupamiento se realiza a partir de grafos no dirigidos con pesos, comenzando con una matriz de afinidad donde los pesos de conexión entre vértices es el valor de correlación entre cada SPN, iniciando en un nodo aleatorio. En cada iteración conectan los nodos restantes y eligen los nodos más cercanos al central obteniendo una nueva matriz de afinidad en cada paso. El algoritmo se detiene cuando el número de nodos más cercanos es menor a un parámetro k. Posteriormente, el grafo es particionado hasta el punto en donde la similitud en un conjunto sea máxima y mínima con respecto a otros conjuntos.

En [Li10b] se realizan agrupamientos mediante campos markovianos aleatorios. Se propone un algoritmo de agrupamiento partiendo de una matriz que contiene todas las correlaciones entre SPNs de diversas cámaras. En cada iteración el algoritmo agrupa dentro de clases los SPN más similares haciendo uso de las características locales de los campos markovianos aleatorios y asigna una nueva etiqueta de clase a cada SPN maximizando una función de probabilidad. El criterio para detener el algoritmo se cumple cuando no hay cambios en las etiquetas después de cierto número de iteraciones.

El algoritmo propuesto en [CAPI10] utiliza agrupamiento jerárquico para agrupar las imágenes. Previo al algoritmo de agrupamiento, los autores aplican una función de mejora del ruido del sensor, que fortalece los componentes bajos y atenúa los componentes altos en el dominio wavelet, con la finalidad de eliminar los detalles de la escena en el mismo. Con una matriz de similitud que contiene todas las correlaciones entre los diferentes SPN y tomando como punto de partida a cada imagen como un grupo único, el algoritmo agrupa los dos grupos con un valor de correlación más alta formando un solo grupo y actualiza la matriz con una nueva fila y columna que vienen a sustituir las filas y columnas de los grupos agrupados. El criterio de enlace elegido para mezclar dos grupos fue el de enlace promedio. En cada iteración del algoritmo se almacena en una partición el estado de los grupos en ese momento y se calcula el coeficiente silueta global. Al final del algoritmo se elige la partición cuyo valor del coeficiente silueta sea mínimo. El número de grupos en ese punto debería corresponder al número de dispositivos que existen inicialmente, así como el contenido de cada grupo a los SPN de cada dispositivo. Los autores realizan una etapa de entrenamiento con el algoritmo descrito y una etapa de clasificación para las imágenes restantes. Para ello basta obtener el promedio de los SPN por cada grupo y compararlos

contra las imágenes restantes; la imagen se clasificará dentro del grupo cuya correlación sea más alta.

10.3 Ataques al Análisis Forense de Imágenes

En comparación con el destacado papel de las imágenes digitales en la sociedad multimedia de hoy en día, la investigación en el campo de la autenticidad de la imagen se encuentra todavía en una fase muy preliminar. La mayoría de las publicaciones en este campo emergente todavía carecen de discusiones rigurosas y robustas contra los falsificadores estratégicos [GKWB07].

El área que se encarga de estudiar ataques a las técnicas de análisis forense de imágenes es conocida como contramedidas forenses (del inglés *counter-forensics*). Los ataques contra los algoritmos forenses de imágenes digitales son aquellas técnicas cuyo objetivo es confundir sistemáticamente a los procedimientos de identificación de la fuente de la imagen o de detección de manipulaciones maliciosas en las imágenes. Estos ataques pueden tener uno de los siguientes objetivos: camuflaje de post-procesamientos maliciosos sobre la imagen o manipulación de la identificación de la fuente (destrucción o falsificación). Un esquema de esta clasificación se aprecia en la Figura 10.3.

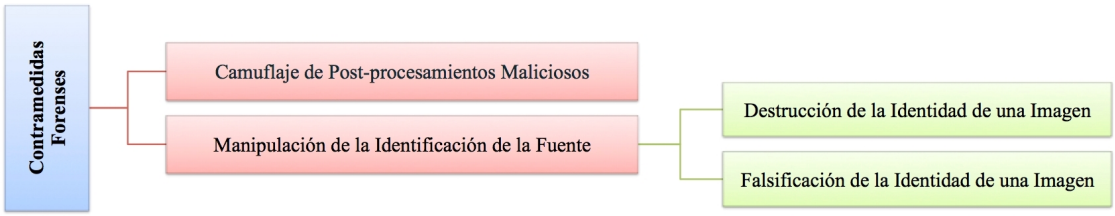


Figura 10.3: Clasificación de las contramedidas forenses

10.3.1 El Camuflaje de Post-Procesamientos

Estas técnicas tienen como objetivo ocultar la existencia de algún proceso aplicado a una imagen analizando los rasgos que éstos dejan sobre la imagen durante su aplicación para así poder contrarrestarlos. En [PF05] se estudian las dependencias introducidas durante el re-dimensionamiento o la rotación de las imágenes. En [LF03] se estudian los coeficientes estadísticos de los JPEG para detectar la re-compresión. En [CSS07] se analiza la fase de congruencia para detectar la composición de imágenes a través del recortado y pegado de diferentes imágenes.

En [GKWB07] se presenta una propuesta para ocultar el proceso de re-muestreo. El re-muestreo es el redimensionamiento con interpolación de las imágenes. Este proceso es muy común en las operaciones primitivas de imágenes como escalamiento y rotación. Los algoritmos detectores de re-muestreo se basan en la búsqueda de las dependencias sistemáticas y periódicas entre píxeles vecinos insertadas cuando se aplica la operación de

re-muestreo. Para ocultar el re-muestreo es necesario romper las equidistancias periódicas introduciendo distorsiones geométricas conocidas como ataques de marca de agua. En este caso se superpone un vector de distorsión aleatoria a las posiciones de cada píxel donde un parámetro determina el grado de distorsión introducido. Para evitar generar características visibles en la imagen como ruido se debe modular la fuerza de la distorsión empleando dos detectores de bordes: uno en dirección vertical y otro en dirección horizontal.

10.3.2 Manipulación de la Identificación de la Fuente

Así como para el proceso de identificación de la fuente se usa la extracción del ruido del sensor en la imagen, un contraataque lógico para esta técnica consta de la eliminación del ruido del sensor. Dando un paso más se puede pensar también en la posibilidad de eliminar el ruido del sensor de la imagen y sustituirlo por el ruido del sensor que pertenezca a otra cámara.

10.3.2.1 Destrucción de la Identidad de una Imagen

En [GKWB07] se demuestra que la diferencia de las características del dominio wavelet de las imágenes no es suficiente para eliminar el ruido de una imagen. Además, este procedimiento deja rastros visibles sobre la imagen. Existe otro método bastante conocido para la eliminación del ruido de una imagen llamado corrección de sensibilidad. Este método es usado típicamente en astronomía o en el proceso de escaneado de planos para mejorar la calidad de las imágenes. La corrección de sensibilidad se realiza en base a los principales componentes del ruido de la imagen: el ruido de patrón fijo o FPN y el ruido de respuesta no uniforme o PRNU. El ruido FPN se calcula en términos de un marco oscuro d promediando K imágenes x_{oscura} capturadas en un ambiente completamente oscuro que se puede emular cubriendo completamente la lente de la cámara.

El ruido PRNU se calcula en términos de un marco plano f promediando L imágenes $x_{iluminada}$ de una escena iluminada homogéneamente. A las L imágenes se les elimina el ruido FPN mediante la resta del marco oscuro d antes de promediarlas.

Como se describe en [LFG06] [GKWB07], los atacantes pueden intentar evitar la identificación correcta de la fuente, ya que existe la posibilidad de eliminar y extraer la huella de una imagen. La destrucción de la huella de una imagen x generada con una cámara específica se realiza con la Ecuación 10.1 restando a la imagen original x el marco oscuro d y dividiendo el resultado de la diferencia por el marco plano f.

$$\tilde{x} = \frac{x - d}{f} \tag{10.1}$$

A pesar de que los resultados obtenidos con esta técnica son buenos, hay algunos inconvenientes:

- Llevar a cabo una perfecta corrección de sensibilidad en un gran número de fotos es difícil ya que los parámetros para calcular el PRNU y el FPN deben coincidir con los de la imagen a atacar.

- En la propuesta se asume que el atacante puede tener acceso a la cámara fuente de la imagen x para generar los marcos oscuros y planos y éste no es un escenario próximo a la realidad.

Existen otras posibilidades menos robustas para destruir la identidad que en ciertos casos podrían ser efectivas ya que no necesitan contar con imágenes procedentes de la cámara origen para generar el marco oscuro y el marco plano, pero a cambio de esta facilidad la calidad de la imagen puede verse reducida y podrían introducirse algunos rasgos visuales. Por ejemplo, es posible rotar la imagen unos pocos grados, escalar la imagen o aplicar un filtro de desenfoque gaussiano.

10.3.2.2 Falsificación de la Identidad de una Imagen

De igual forma que se puede eliminar el ruido en una imagen haciendo uso de la técnica de corrección de sensibilidad, se puede inyectar el ruido de la imagen de otra cámara diferente mediante la corrección de sensibilidad inversa con la Ecuación 10.2 [GKWB07]:

$$\tilde{y} = \tilde{x} \cdot f_{falsa} + d_{falsa} \tag{10.2}$$

donde f_{falsa} y d_{falsa} corresponden a la cámara que se pretende plagiar y $\tilde{x}$ es la imagen original sin ruido.

En [SLFK10] se propone el Algoritmo 8 para falsificar la identidad de una cámara.

Algoritmo 8: Falsificación de la identidad de una cámara

1. Calcular el promedio de las huellas $F(C1)$ de la cámara $C1$ con la que se atacará;
2. Tomar una fotografía P con la segunda cámara $C2$;
3. Sumar $F(C1)$ a la fotografía P;

En el caso de que las dimensiones de $F(C1)$ y P no coincidan, es necesario aplicar un recorte o una reconstrucción para igualar el tamaño de las imágenes.

También se propone una mejora al algoritmo de falsificación anterior para enmascarar los rasgos de la cámara $C2$. Esta técnica se presenta en el Algoritmo 9.

Algoritmo 9: Falsificación de la identidad de una cámara para imágenes con dimensiones diferentes

1. Calcular el promedio de las huellas $F(C1)$ de la cámara $C1$ con la que se atacará;
2. Calcular el promedio de las huellas $F(C2)$ de la cámara $C2$;
3. Sumar $F(C1)$ a la fotografía P;
4. Tomar una fotografía P con la cámara $C2$;
5. Restar $F(C2)$ a P;

Al restar $F(C2)$ se trata de eliminar la correlación entre la fotografía P y la cámara $C2$.

10.4 Síntesis del Capítulo

En este capítulo se ha revisado el trabajo relacionado con las dos líneas de investigación principales de esta tesis: la identificación de la fuente de adquisición de imágenes digitales tanto en escenarios abiertos como cerrados, así como los posibles ataques que las técnicas de identificación de la fuente pueden sufrir.

Capítulo 11

Identificación de la Fuente de Adquisición de Imágenes Basada en el Uso de Características de la Imagen

La identificación del tipo de dispositivo o la marca y el modelo de la fuente de la imagen son dos ramas importantes del análisis forense de las imágenes digitales. En este capítulo se abordan ambas con un enfoque basado en la extracción de características del contenido de las imágenes y la clasificación mediante máquinas de soporte vectorial.

Este capítulo está estructurado como sigue: En primer lugar se presenta la especificación de la técnica propuesta para la identificación de la fuente de adquisición de imágenes (tipo de fuente o marca y modelo de origen) basada en la extracción de características del contenido de la imagen y los diferentes conjuntos de características (Ruido, Color, IQM y Wavelets). A continuación se realiza un conjunto de experimentos para la identificación del tipo de dispositivo y la identificación de la fuente de adquisición de la imagen. El capítulo finaliza con una breve síntesis de lo expuesto en el mismo.

11.1 Generalidades

Como se ha observado en los trabajos relacionados existe un gran variedad de características clasificadas en tipos según su base de obtención. Un algoritmo de identificación de la fuente basado en características no obtiene mejores resultados únicamente por tener mayor número de ellas. De hecho puede darse el caso de que al utilizar un mayor número de características los resultados empeoren. La base para que un algoritmo obtenga buenos resultados es una conjunción de muchos factores entre los que destacan la elección de características que realmente determinen la identidad de la imagen, la elección de un número de características adecuado y la elección de un buen método de clasificación. Desgraciadamente, muchas veces sólo la experimentación con imágenes reales puede ofrecer datos sobre los resultados de un algoritmo basado en las características.

La técnica propuesta para la identificación de la fuente de adquisición de una imagen (tipo de fuente o marca y modelo de la fuente) está basada en la extracción de características del contenido de la propia imagen.

La contribución principal a esta técnica es un nuevo enfoque de generación de características en el que se combinan el patrón del ruido del sensor, las características de color, las métricas de calidad de la imagen y las características wavelets. La combinación de estas características permite la identificación de imágenes de diferentes tipos de dispositivos (imágenes de dispositivos móviles, imágenes obtenidas de un escáner e imágenes generadas por computador) y la identificación de marca y modelo en imágenes de dispositivos móviles.

Dependiendo de la naturaleza de la obtención de las mismas, el conjunto de características a utilizar puede clasificarse en cuatro grandes grupos:

- Características del ruido (16 características).
- Características del color (12 características).
- IQM (40 características).
- Wavelets (81 características).

Para la identificación se utilizan 2 conjuntos de imágenes: imágenes de fuentes conocidas para entrenar el clasificador SVM [HCL03] y otro conjunto de imágenes de fuentes desconocidas que se utilizarán en la fase de predicción para averiguar su fuente de adquisición.

Sobre la parte de la clasificación en [MST94] se realiza un estudio de diferentes métodos de clasificación como pueden ser los clasificadores basados en distancias, los clasificadores bayesianos, redes neuronales, los algoritmos de agrupamiento y los clasificadores SVM. Como puede verse en el estado del arte, el uso de clasificadores SVM es ampliamente utilizado para estos menesteres. El núcleo utilizado para clasificar es *Non-linear RBF*, dado que es recomendado cuando no se cuenta con información a priori de los datos.

Los parámetros utilizados en el clasificador SVM son los mismos que los utilizados en [RCAGSO$^+$13]. Existen muchas implementaciones de clasificadores SVM, concretamente en este trabajo se optó por utilizar la librería LibSVM [CL13].

A continuación se va a realizar un análisis detallado de cada uno de los conjuntos de características citados anteriormente.

11.1.1 Características del Ruido

El proceso de generación de imágenes suele introducir en ellas varios defectos, los cuales crean ruido que aparece en la imagen final. Un tipo de ruido es causado por defectos de la matriz CFA, entre los cuales se incluyen los defectos de puntos calientes, los píxeles muertos, las trampas de píxeles, los defectos de columna y los defectos de grupo. Estos defectos causan que dichos píxeles difieran en gran medida de los restantes de la imagen original, siendo en muchos casos indiferente que se tenga una u otra imagen, ya que este píxel mostrará siempre el mismo valor. Por ejemplo, los píxeles muertos aparecerán en la imagen como píxeles negros o los puntos calientes aparecerán como píxeles muy

brillantes. El patrón de ruido en una imagen se refiere a cualquier patrón espacial que no cambia de una imagen a otra y es causado por una corriente de oscuridad y el PRNU [KMC+07]. Existen distintos filtros para conseguir suavizar el efecto de este ruido, si bien es el Gaussiano el más utilizado. Su núcleo es separable, permitiendo velocidad de cómputo; elimina los componentes de "alta frecuencia" de la imagen al ser un filtro paso bajo que elimina mejor el ruido mientras reduce la borrosidad en los bordes; garantiza resultados no negativos (siendo siempre otra imagen válida); el grado de suavizado es controlado por el parámetro σ (a mayor σ, una suavización más intensa); finalmente, produce un suavizado más uniforme que otros filtros como el filtro de media. Estas propiedades hacen que sea el filtro usado para eliminar el ruido en las imágenes y posteriormente para obtener las distintas características.

Uno de los objetivos de esta técnica es conseguir un conjunto de características que permita diferenciar entre los distintos tipos de dispositivos. Para ello, como un primer paso, se tiene en cuenta que las cámaras digitales utilizan un conjunto bidimensional de sensores mientras que la mayoría de los escáneres utilizan un conjunto lineal de sensores. En el caso de los escáneres la misma disposición lineal del sensor se traslada para generar toda la imagen. Por tanto, se espera encontrar periodicidad del ruido del sensor entre las filas de la imagen escaneada. Asimismo, no hay razón para encontrar una periodicidad del ruido del sensor entre las columnas de la imagen escaneada. Para el caso de cámaras digitales este tipo de periodicidad del ruido no existe. Esta diferencia se puede utilizar como base para discriminar entre los distintos tipos de dispositivos. La extracción de las características del ruido se basa en [KMD09].

Sea I una imagen de $M \times N$ píxeles, siendo M las filas y N las columnas. Sea I_{ruido} el ruido correspondiente a la imagen original e $I_{sinruido}$ la imagen sin ruido. El ruido de la imagen original se obtiene, por tanto, usando la Ecuación 11.1:

$$I_{ruido} = I - I_{sinruido} \tag{11.1}$$

Después se restará cada componente de color de la imagen sin ruido a cada componente de color de la imagen original, obteniéndose los componentes de ruido de cada píxel desglosados por componente de color.

El ruido de la imagen original I_{ruido} puede ser modelado como la suma de dos componentes, el ruido constante $I_{ruidoconstante}$ y el ruido aleatorio $I_{ruidoaleatorio}$. Para los escáneres el ruido constante sólo depende del índice de la columna, ya que el mismo sensor es trasladado verticalmente para generar la imagen completa. La media del ruido de todas las columnas $\hat{I}_{ruidoconstante}(1, j)$ (Ecuación 11.2) puede ser usada como patrón de referencia ya que las componentes aleatorias del ruido se anularán.

$$\hat{I}_{ruidoconstante}(1, j) = \frac{\overset{M}{\underset{i=1}{}} I_{ruido}(i, j)}{M}, 1 \leq j \leq N \tag{11.2}$$

Siendo la correlación normalizada una de las medidas más comúnmente utilizadas en la identificación de la fuente de imagen [Blo08] [Fri09] [Li10a] [CAPI10], para detectar la

similitud entre las diferentes filas con el patrón de referencia, se utiliza la correlación de éstas con dicho patrón mediante la Ecuación 11.3.

$$correlación\,(X,Y) = \frac{X - \overline{X} \;\cdot\; Y - \overline{Y}}{X - \overline{X} \;\cdot\; Y - \overline{Y}} \tag{11.3}$$

Posteriormente, se realiza el mismo proceso para detectar la similitud de las columnas con el patrón de referencia. Finalmente, tras obtener las correlaciones entre las filas y entre las columnas se obtienen el conjunto de características en sí. Cabe destacar a la hora de obtener las características, que la orientación de la imagen en el caso de los escáneres es fundamental, ya que dependiendo de ésta las características obtenidas son completamente diferentes.

Para cada tipo de correlación se obtienen valores estadísticos de primer orden: media, mediana, máximo y mínimo. Se descarta como característica la moda, ya que es una característica inútil, dado que al tratarse de valores flotantes, no existen en la inmensa mayoría de casos valores repetidos. Además, al realizar pruebas truncando los valores flotantes, los resultados no son buenos, disminuyendo los porcentajes de aciertos. Otras características de orden alto obtenidas son la varianza, la curtosis y la asimetría. Todas ellas miden valores estadísticos más específicos que las anteriores. Asimismo, se añaden las características de la razón entre las correlaciones de filas y de columnas. Por último, se incluye la característica del ruido medio del píxel. Esta característica no depende de las correlaciones de filas o columnas con el patrón de referencia, sino que es independiente y permite distinguir entre los distintos tipos de dispositivos, como pueden ser las imágenes generadas por computador.

En total, se obtienen un conjunto de 16 características: 7 características de filas, 7 de columnas, la razón entre las correlaciones de filas y de columnas y el ruido medio del píxel.

11.1.2 Características de Color

La configuración de los filtros de la matriz CFA, el algoritmo de interpolación cromática y la técnicas aplicadas al procesamiento del color hacen que las señales contenidas en la bandas de color tengan tratamientos y patrones específicos. Con el objetivo de determinar las diferencias en las características del color para los diferentes modelos de cámaras es necesario examinar las estadísticas de primer y segundo orden de las imágenes tomadas con ellas. A continuación se proponen un conjunto de 12 características de color basadas en las de [HLZ10].

- **Valor medio de los píxeles**: Para esta medida se asume que el promedio de los valores de los canales RGB de una imagen debe dar como resultado el color gris, siempre y cuando la imagen tenga suficientes variaciones de color. Esta medida se realiza para cada uno de los canales RGB (3 características).

- **Correlación de los pares RGB**: Con esta medida se expresa el hecho de que dependiendo de la estructura de la cámara, la correlación entre las diferentes bandas de color puede variar. En la implementación de esta característica se utiliza el

coeficiente de correlación de Pearson para determinar la correlación en los valores de cada una de las bandas. Como resultado se obtienen 3 características que provienen de medir la correlación entre las bandas RG, RB y GB.

- **Distribución de vecindad del centro de masa**: Esta medida se calcula para cada banda por separado. Primero se calcula el número total de píxeles para cada valor de color, resultando un vector de 256 componentes. Después, con estos valores calculados se obtienen la suma de los valores vecinos, es decir, para cada valor i del vector anteriormente calculado se suma la componente $i-1$ e $i+1$. Por último, se calcula el centro de masa de este último vector, lo que va a devolver un valor entre 0 y 255 (3 características).

- **Razón de la energía entre los pares RGB**: Esta característica depende del proceso de corrección de puntos blancos que realiza la cámara. Son 3 características que están definidas por la Ecuación 11.4.

$$E_1 = \frac{|G|^2}{|B|^2} \qquad E_2 = \frac{|G|^2}{|R|^2} \qquad E_3 = \frac{|B|^2}{|R|^2} \tag{11.4}$$

11.1.3 Métricas de Calidad de la Imagen

Los diferentes modelos de cámaras producen imágenes de diferente calidad. Puede haber diferencias en la luminosidad de la imagen, la nitidez o en la calidad del color. Estas diferencias hacen que se proponga un conjunto de métricas de calidad como características que ayudan a diferenciar la fuente de las imágenes. Las métricas de calidad de la imagen son de suma importancia para proporcionar datos cuantitativos sobre la calidad de una imagen *renderizada* [ASS02] [AMS03].

Con el fin de obtener una diferencia más detallada de las imágenes, existen diferentes categorías para estas métricas: las medidas basadas en las diferencias de los píxeles, las medidas basadas en la correlación y las medidas basadas en la distancia espectral.

Para obtener este conjunto de métricas se necesita la imagen original y una imagen filtrada en la que se reduzca el ruido de la imagen original. Por los motivos mencionados en la Sección 11.1.1, se utiliza un filtrado gaussiano que permite llevar a cabo el suavizado de la imagen. Para la obtención del núcleo gaussiano bidimensional se utiliza la Fórmula 11.5.

$$2\pi\sigma^{2\ -1} * e^{\frac{-\left(i^2+j^2\right)}{2\sigma^2}} \tag{11.5}$$

donde i es la distancia desde el origen en el eje horizontal, j es la distancia desde el origen en el eje vertical (por ejemplo, en el centro de la matriz i y j son iguales a 0) y σ es la desviación estándar de la Distribución Gaussiana que representa un valor umbral o factor especificado por el usuario.

Una vez obtenido el núcleo se normaliza para que la suma de todas sus componentes sea 1. Esto es necesario para obtener una imagen suavizada pero con los mismos colores que la original. La normalización se realiza dividiendo cada componente entre la suma de

los valores de todas las componentes.

Para obtener las métricas se utiliza un filtro gaussiano con un núcleo de tamaño 3×3 y $\sigma = 0.5$. En la Ecuación 11.6 se muestra el filtro resultante.

$$h = \begin{matrix} 0.026625868 & 0.196748265 & 0.026625868 \\ 0.196748265 & 1.453735842 & 0.196748265 \\ 0.026625868 & 0.196748265 & 0.026625868 \end{matrix} \tag{11.6}$$

Cada píxel de la nueva imagen se calcula realizando la transformación de vecindad mediante la Ecuación 11.7 sobre el píxel de la imagen original utilizando el núcleo resultante de aplicar el filtro anterior.

$$\begin{aligned} I'(x,y) = {} & h(0,0) * I(x-1,y-1) + h(0,1) * I(x,y-1) + h(0,2) * I(x+1,y-1) + \\ & h(1,0) * I(x-1,y) + h(1,1) * I(x,y) + h(1,2) * I(x+1,y) + \\ & h(2,0) * I(x-1,y+1) + h(2,1) * I(x,y+1) + h(2,2) * I(x+1,y+1) \end{aligned} \tag{11.7}$$

Es necesario tener en cuenta los bordes de la imagen al realizar la transformación. En este caso se ha optado por considerar un borde exterior con píxeles de valor 0.

Las medidas basadas en las diferencias de los píxeles calculan la distorsión entre dos imágenes en base a sus diferencias entre píxeles. Entre estas medidas se encuentran la Métrica de Minkowsky, el Error Absoluto Medio y el Error Cuadrático Medio (Ecuaciones 11.9 a 11.11, respectivamente).

Las medidas basadas en la correlación estiman la similitud entre dos imágenes digitales en términos de la función de correlación y son complementarias a las medidas basadas en las diferencias de píxeles. En esta categoría están la Distancia Czekonowsky, la Correlación Cruzada Normalizada y el Contenido Estructural (Ecuaciones 11.8, 11.14 y 11.15, respectivamente).

Las medidas de distancia espectral consideran las funciones de penalización de distorsión obtenidas a partir del complejo espectro de Fourier de las imágenes. Las medidas agrupadas en esta categoría son: la Fase Espectral, la Magnitud Espectral, la Distancia Espectral Ponderada, la Mediana del Bloque de Magnitud Espectral, la Mediana del Bloque de Fase Espectral y la Mediana del Bloque de Distancia Espectral Ponderada (Ecuaciones 11.20 a 11.22 y 11.26 a 11.28, respectivamente).

A continuación se muestra la especificación de las 40 características IQM utilizadas basadas en [HLZ10].

- **Distancia Czekonowsky**: Es una métrica útil para comparar vectores con componentes no negativas como es el caso de las imágenes en color y se calcula con la Ecuación 11.8.

$$M = \frac{1}{MN} \sum_{i=1}^{M} \sum_{j=1}^{N} \left(1 - \frac{2 \sum_{k=1}^{3} min \left(C_k(i,j), \hat{C}_k(i,j) \right)}{\sum_{k=1}^{3} C_k(i,j) + \hat{C}_k(i,j)} \right) \tag{11.8}$$

En esta ecuación y en las posteriores $C_k(i,j)$ y $\hat{C}(i,j)$ se refieren a los píxeles en la posición (m,n) de la imagen original y la imagen suavizada (imagen filtrada con reducción de ruido), respectivamente. Asimismo, M y N son el tamaño horizontal y vertical de la imagen, respectivamente.

- **Métricas de Minkowsky**: Para $\gamma = 1$ y $\gamma = 2$ se basan en la Ecuación 11.9.

$$M_\gamma = \frac{1}{K} \sum_{k=1}^{K} \left\{ \frac{1}{N^2} \sum_{i,j=1}^{N} C_k(i,j) - \hat{C}_k(i,j)^{\gamma} \right\}^{\frac{1}{\gamma}} \tag{11.9}$$

Esta ecuación calcula la norma L_γ de disimilitud entre dos imágenes, donde N^2 es el número total de píxeles. En esta ecuación (y en adelante), k se refiere a cada uno de los canales de la imagen. Hay que tener en cuenta que esta ecuación realiza el promedio de la métrica de Minkowsky para todos los canales de la imagen.

$\gamma = 1$ se corresponde con el *Error Absoluto Medio* (EAM) y $\gamma = 2$ con el *Error Cuadrático Medio* (ECM) (Ecuaciones 11.10 y 11.11, respectivamente). En ambos casos, valores elevados de EAM o ECM se corresponden con imágenes de baja calidad.

  - *Error Absoluto Medio*:

$$EAM = \frac{1}{MN} \sum_{m=1}^{M} \sum_{n=1}^{N} C_k(i,j) - \hat{C}_k(i,j) \tag{11.10}$$

  - *Error Cuadrático Medio*:

$$ECM = \frac{1}{MN} \sum_{m=1}^{M} \sum_{n=1}^{N} C_k(i,j) - \hat{C}_k(i,j)^{2} \tag{11.11}$$

Estas métricas se aplican a cada una de las bandas por separado, por lo que se obtienen tres características para el EAM y otras tres para el ECM.

- **Error Cuadrático Medio Laplaciano**: Está basado en la importancia de la medición de los bordes y se define por la Ecuación 4.13. Un valor alto de este error (del inglés LMSE) indica que la imagen es de baja calidad. Se define de la siguiente

forma:

$$LMSE = \frac{\sum_{m=1}^{M}\sum_{n=1}^{N}\left[L\left(x\left(m,n\right)\right) - L\quad x^{(m,n)}\right)\right]^2}{\sum_{m=1}^{M}\sum_{n=1}^{N}\left[L\left(x\left(m,n\right)\right)\right]^2} \tag{11.12}$$

donde $L\left(x\left(m,n\right)\right)$ es el operador Laplaciano y se estima por la Ecuación 11.13.

$$\begin{aligned} L\left(x\left(m,n\right)\right) &= x\left(m+1,n\right) + x\left(m-1,n\right) + x\left(m,n+1\right) + \\ &\quad x\left(m,n-1\right) - 4x\left(m,n\right) \end{aligned} \tag{11.13}$$

- **Correlación Cruzada Normalizada**: La cercanía entre dos imágenes digitales también puede ser cuantificada en términos de una función de correlación. La métrica de calidad de la medida de correlación cruzada normalizada para cada banda de la imagen k se define a través de la Ecuación 11.14.

$$NCC = \frac{\sum_{i,j=0}^{N-1} C_k\left(i,j\right) * \hat{C}_k\left(i,j\right)}{\sum_{i,j=0}^{N-1} C_k\left(i,j\right)^2} \tag{11.14}$$

- **Contenido Estructural**: La métrica de calidad del contenido estructural de una imagen está definida para cada banda k por la Ecuación 11.15.

$$SC = \frac{\sum_{i,j=0}^{N-1} C_k\left(i,j\right)^2}{\sum_{i,j=0}^{N-1} \hat{C}_k\left(i,j\right)^2} \tag{11.15}$$

- **Medidas Espectrales**: Para determinar estas medidas se calcula la DFT de la imagen original y de la suavizada, denotadas como $\tau_k\left(u,v\right)$ y $\hat{\tau}_k\left(u,v\right)$ para una banda k (Ecuaciones 11.16 y 11.17, respectivamente).

$$\tau_k\left(u,v\right) = \sum_{m,n=0}^{N-1} C_k\left(m,n\right) * e^{\left[-2\pi i m \frac{u}{N}\right]} * e^{\left[-2\pi i n \frac{v}{N}\right]} \tag{11.16}$$

$$\hat{\tau}_k\left(u,v\right) = \sum_{m,n=0}^{N-1} \hat{C}_k\left(m,n\right) * e^{\left[-2\pi i m \frac{u}{N}\right]} * e^{\left[-2\pi i n \frac{v}{N}\right]} \tag{11.17}$$

donde $u = 0,\ldots,M-1$, $v = 0,\ldots,N-1$ y (u,v) son las coordenadas de los píxeles de la imagen en el dominio de la transformada. La fase y magnitud del espectro de

la DFT se definen por las Ecuaciones 11.18 y 11.19, respectivamente.

$$\varphi(u,v) = \arctan(\tau_k(u,v)) \tag{11.18}$$

$$M(u,v) = |\tau_k(u,v)| \tag{11.19}$$

Con los conceptos anteriores se pueden definir las siguientes métricas de calidad para cada banda de la imagen: la Fase Espectral, la Magnitud Espectral y una media ponderada entre la fase y la magnitud espectral (Ecuaciones 11.20 a 11.22).

- *Fase Espectral*:

$$SP = \frac{1}{MN} \sum_{u=1}^{M} \sum_{v=1}^{N} |\varphi(u,v) - \hat{\varphi}(u,v)|^2 \tag{11.20}$$

- *Magnitud Espectral*:

$$SM = \frac{1}{MN} \sum_{u=1}^{M} \sum_{v=1}^{N} M(u,v) - \hat{M}(u,v)^2 \tag{11.21}$$

- *Distancia Espectral Ponderada*: Realiza una media ponderada entre la fase y la magnitud espectral.

$$WSD = \rho * SM + (1-\rho) * SP \tag{11.22}$$

  donde para este caso $\rho = 2,5 * 10^{-5}$.

Estas características también se pueden obtener para cada bloque de la imagen. Por tanto, se divide la imagen en L bloques de tamaño $b \times b$ y luego se calculan dichas características. De esta manera se pueden definir las características definidas por las Ecuaciones 11.23, 11.24 y 11.25 sobre el bloque l-ésimo para cada banda del bloque:

$$J^l{}_{\varphi} = \left(\sum_{u,v=0}^{b-1} \varphi^l(u,v) - \hat{\varphi}^l(u,v)^{\gamma} \right)^{\frac{1}{\gamma}} \tag{11.23}$$

$$J^l{}_{M} = \left(\sum_{u,v=0}^{b-1} M^l(u,v) - \hat{M}^l(u,v)^{\gamma} \right)^{\frac{1}{\gamma}} \tag{11.24}$$

$$J^l = \rho * J^l{}_{M} + (1-\rho) * J^l{}_{\varphi} \tag{11.25}$$

Para calcular estas características se ha utilizado $\gamma = 2$ y un tamaño de bloque de 32×32.

Posteriormente, para cada bloque se obtienen las siguientes características: la Mediana de Bloque de la Magnitud Espectral, la Mediana de Bloque de la Fase

Espectral y la Mediana de Bloque de la Distancia Espectral Ponderada (Ecuaciones 11.26 a 11.28).

- *Mediana de Bloque de la Magnitud Espectral*:

$$\text{MBSM} = mediana\ J^l{}_M,\ \text{l} = 1,\ ...,\ \text{L} \tag{11.26}$$

- *Mediana de Bloque de Fase Espectral*:

$$\text{MBSP} = mediana\ J^l{}_\varphi,\ \text{l} = 1,\ ...,\ \text{L} \tag{11.27}$$

- *Mediana de Bloque de la Distancia Espectral Ponderada*:

$$\text{MBWSM} = mediana\ J^l,\ \text{l} = 1,\ ...,\ \text{L} \tag{11.28}$$

- **Medidas basadas en el sistema visual humano**: Las imágenes pueden ser procesadas mediante filtros que simulan el HVS. Uno de los modelos utilizados para ello es un filtro pasa banda con una función de transferencia en coordenadas polares definido por la Ecuación 11.29.

$$H(\rho) = \left\{ \begin{array}{c} 0.05e^{\rho^{0.554}} \quad \rho < 7 \\ e^{-9[|log_{10}\rho - log_{10}9|]^{2.3}} \quad \rho \geq 7 \end{array} \right\} \tag{11.29}$$

donde $\rho = \quad \overline{(u^2 + v^2)}$. Se define el operador U por la Ecuación 11.30.

$$U\left\{C\left(i,j\right)\right\} = DCT^{-1} \quad H \qquad \overline{u^2 + v^2} \quad \omega\left(u,v\right) \tag{11.30}$$

donde $\omega\left(u,v\right)$ denota la transformada discreta del coseno bidimensional DCT de la imagen y DCT^{-1} es la DCT inversa bidimensional.

Finalmente, las métricas de calidad que se obtienen para cada banda de la imagen basadas en estas medidas son: Error Absoluto Normalizado y HVS basado en L2 (Ecuaciones 11.31 y 11.32, respectivamente).

$$NAE = \frac{\sum_{i,j=0}^{N-1} U\left\{C_k\left(i,j\right)\right\} - U \quad \hat{C}_k\left(i,j\right)}{\sum_{i,j=0}^{N-1} \left|U\left\{C_k\left(i,j\right)\right\}\right|} \tag{11.31}$$

$$L2 = \left\{ \frac{1}{N^2} \sum_{i,j=0}^{N-1} U\left\{C_k\left(i,j\right)\right\} - U \quad \hat{C}_k\left(i,j\right) \quad ^2 \right\}^{\frac{1}{2}} \tag{11.32}$$

El HVS es demasiado complejo para ser plenamente comprendido y es intrigante aprender el papel de las medidas basadas en el HVS. Sin embargo, la incorporación de un

modelo simplificado de HVS en medidas objetivas conduce a mejores correlaciones [Nil85] [NB92] [Wat93] [FBA97], de ahí su importancia.

11.1.4 Características Wavelet

Debido a la propiedad determinista del patrón de ruido del sensor presente en una imagen, este patrón puede usarse como huella para identificar el dispositivo que generó la imagen. Haciendo una analogía, se puede decir que el patrón de ruido del sensor es para una cámara digital lo que la huella dactilar para un ser humano. Así, para clasificar e identificar la fuente de adquisición de una imagen se requiere de un algoritmo que nos permita extraer el ruido del sensor y otro que nos permita obtener las características de las huellas obtenidas.

En contraste a la Transformada de Fourier que representa las señales como una suma de ondas senoidales que no están localizadas en el tiempo y espacio, la Transformada Wavelet es más conveniente en el análisis de señales con cambios abruptos como son las imágenes ya que sus funciones están ubicadas en tiempo y espacio. Aunado a lo anterior entre las diferentes familias de funciones wavelets (Haar, Daubechies, Coiflet, Symlet, Meyer, etc.) existe una gran cantidad de alternativas para analizar las señales permitiendo elegir la base de funciones cuya forma se aproxime mejor a las características de la señal que se desea analizar. En base a los resultados de trabajos anteriores [RCAGSO$^+$13] [SOGVAG$^+$15] las funciones Daubechies son las que mejores resultados han tenido en la extracción del ruido del sensor.

La Transformada Wavelet permite separar la información a manera de frecuencias; de esta forma se puede analizar y/o modificar sólo la información de las frecuencias que son de algún interés en particular como el ruido del sensor para así poder extraer sus características.

Para la extracción del ruido del sensor se utiliza el algoritmo presentado en [AG15].

Finalmente, con el Algoritmo 10 se calculan un total de 81 características (3 canales $\times$ 3 componentes wavelets $\times$ 9 momentos centrales).

Algoritmo 10: Extracción de características

Input: Imagen
Huella del sensor de la imagen
Result: 81 características

1. **procedure** ExtraerCaracteristicas(I)
2. Separar los canales R, G y B de la huella del sensor;
3. **foreach** *canal de color* **do**
4. Hacer una descomposición wavelet de un nivel;
5. **foreach** $c \in \{H,V,D\}$ **do**
6. Calcular k momentos centrales con $m_k = \frac{1}{n} \sum_{i=1}^{n} |c_i - \overline{c}|^k$;
7. **end procedure**

11.2 Experimentos

Para la evaluación de la propuesta se realizaron dos tipos de experimentos: identificación del tipo de dispositivo fuente e identificación de la fuente de adquisición en dispositivos móviles.

La clasificación de las imágenes se realiza sobre lo que puede denominarse un *conjunto cerrado* de elementos. Es decir, las clases de los elementos utilizados en el entrenamiento son las mismas clases que las utilizadas en la predicción. Las imágenes utilizadas en la fase de entrenamiento son diferentes a las utilizadas en la fase de predicción. Los experimentos se realizan sobre 200 imágenes de cada tipo de dispositivo, 100 para la fase de entrenamiento y 100 para la fase de predicción. Todas las imágenes tienen una resolución mayor a 1024×768. No existe ninguna restricción sobre el contenido de la imagen o los parámetros de configuración de la cámara.

11.2.1 Identificación del Tipo de Dispositivo Fuente

En el experimento se utilizaron imágenes de 12 teléfonos móviles, imágenes obtenidas de 15 escáneres e imágenes generadas digitalmente por un computador.

Las imágenes de teléfonos móviles han sido obtenidas de teléfonos propios conocidos. Por tanto, se puede asegurar la originalidad de la fuente; algunos de ellos del mismo fabricante.

Las imágenes de escáneres y las generadas por computador se descargaron de Flickr. A las imágenes descargadas de Internet se les aplicaron diversos filtros para obtener un conjunto de mayor fiabilidad y que indujera el menor ruido posible en los experimentos. Las fotos descargadas de Flickr son originales sin ningún tipo de redimensionamiento. Como segundo filtro para las imágenes de escáneres se tomaron las que tenían la etiqueta "*scanned images*" y hacían referencia a un modelo comercial de escáner. De las imágenes generadas por computador se descartaron las imágenes con la etiqueta "Modelo" procedente de una cámara o escáner. A las imágenes generadas por computador no se le puede indicar una información precisa sobre el número de aplicaciones o tipo de computadores utilizadas. Se observa que existe un alto número de clases de dispositivos (marcas y modelos) diferentes de los tres tipos, lo cual dificulta enormemente la clasificación.

En los experimentos se han tenido en cuenta los siguientes parámetros de configuración: tamaño de recorte aplicado, posición del recorte (centrado o esquina superior izquierda) y aplicación de distintos conjuntos de características (ruido, color, IQM y wavelet).

En las Tabla 11.1 se muestran los resultados de las tasas de acierto y los parámetros de configuración utilizados en los 10 experimentos.

Tabla 11.1: Identificación del tipo de dispositivo fuente (cámara, escáner, computador)

C\ticas	Recorte	Posición de Recorte	Tipo de Dispositivo			% de Acierto
			Cámara	Computador	Escáner	
Ruido	No	-	70	54	57	**59,95**
Ruido	1024×768	Centrado	66	80	46	**62,39**
Ruido	800×600	Centrado	76	60	49	**60,68**
Ruido	640×480	Centrado	62	61	48	**56,62**
Ruido	1024×768	Esquina sup. izq.	76	59	40	**56,40**
Ruido	800×600	Esquina sup. izq.	65	38	44	**47,72**
Ruido	640×480	Esquina sup. izq.	74	54	37	**52,88**
Todas	1024×768	Centrado	66	73	72	**70,26**
Todas	800×600	Centrado	69	74	71	**71,30**
Todas	640×480	Centrado	77	73	63	**70,75**
Promedio			**69,91 %**	**61,35 %**	**51,42 %**	**60,42 %**

Del análisis de los resultados pueden obtenerse conclusiones generales y específicas sobre las distintas configuraciones utilizadas en cada experimento. Englobando todos los experimentos, se observa que las tasas de acierto no son excesivamente altas (un 60,42 % de media y un 71,30 % en el mejor de los casos), por lo cual se puede concluir que esta técnica no es especialmente adecuada para tal fin. Es importante recalcar, como se dijo anteriormente, que el número distinto de marcas y modelos utilizados para este experimento es alto, lo cual, como era previsible, hace que las tasas de acierto bajen. Dicho esto, cabe destacar que este estudio sí aporta resultados interesantes sobre los parámetros de configuración utilizados, ya que entre el mejor y el peor resultado hay una diferencia en la tasa media de acierto del 23,48 %.

En general, puede concluirse que la utilización única de las características del ruido no obtiene buenos resultados para la identificación del tipo de fuente cuando el número de dispositivos a clasificar es alto, ya que la tasa media de acierto de todos los experimentos es del 56,65 %. Dado que los resultados no son buenos, posteriormente se realizarán un conjunto de experimentos que reducirán el número de dispositivos y tipos para observar sus resultados.

Los resultados mejoran notablemente cuando se utilizan todas las características para la identificación del tipo de fuente. Dado el alto número de clases los resultados pueden calificarse como aceptables, ya que la tasa media de acierto para todos los experimentos realizados con la utilización de estas características es del 70,77 %. Asimismo, como se observa en los resultados obtenidos, se concluye que el tamaño de recorte afecta a los resultados: A menor tamaño de recorte de menor tasa de acierto, aun no siendo la diferencias extremadamente significativas. También cabe destacar que con un tamaño de recorte 1024×768 se obtienen mejores resultados que cuando se utiliza todo el tamaño de la imagen, es decir, que a partir de un tamaño de recorte dado los resultados empeoran.

A continuación se van a realizar una serie de experimentos reduciendo considerablemente el número de clases a clasificar, con el objetivo de comprobar el comportamiento cuando se utilizan únicamente las características del ruido, ya que con

éstas son las que se obtuvieron anteriormente peores resultados. Los resultados se muestran en la Tabla 11.2.

Tabla 11.2: Uso de características del ruido en la identificación

Tipo de Dispositivo 1	Tipo de Dispositivo 2	Recorte	Posición de Recorte	% de Acierto
Escáner	Teléfono móvil	1024×768	Centrado	95,79
Escáner	Teléfono móvil	640×480	Centrado	96,16
Escáner	Teléfono móvil	400×300	Centrado	96,73
Computador	Teléfono móvil	1024×768	Centrado	79,96
Computador	Teléfono móvil	640×480	Centrado	79,76
Computador	Teléfono móvil	400×300	Centrado	78,55
Computador	Escáner	1024×768	Centrado	82,87
Computador	Escáner	640×480	Centrado	81,10
Computador	Escáner	400×300	Centrado	80,91

Como puede observarse, la tasa de aciertos sube considerablemente como era de esperar, para situarse en un 85,44 % de media. Cuando el número de tipos de dispositivos se reduce a dos, y como consecuencia se reduce el número de clases, los resultados son aceptables.

La primera conclusión general que se obtiene, corrobora una conclusión anterior, ya que se observa que el tamaño del recorte no afecta significativamente a los resultados. Los mejores resultados obtenidos son los que distinguen entre imágenes de un dispositivo móvil y un escáner con un 96,23 % de tasa media de acierto. El segundo mejor resultado se da con la distinción entre imágenes de un escáner y las generadas por computador con un 81,62 % de tasa media de acierto. El peor de los resultados se obtuvo con la distinción de imágenes generadas por computador y por dispositivos móviles con un 79,42 % de tasa media de acierto. Aun así, cualquiera de los resultados de estos experimentos es notablemente mejor que los de la Tabla 11.1. Por tanto, puede concluirse que, en líneas generales, el uso de las características del ruido para la distinción del tipo de fuente sólo obtiene resultados aceptables cuando el número de clases no es alto.

11.2.2 Identificación de la Fuente de la Imagen en Dispositivos Móviles

Dada la importancia de las imágenes de los dispositivos móviles en la actualidad, a continuación se van a mostrar los experimentos realizados para identificar la fuente de adquisición de imágenes de dispositivos móviles. Es decir, la clasificación de un conjunto de imágenes según la marca y modelo de la cámara que las generó.

En este apartado se utilizó un conjunto de 200 imágenes, 100 para entrenamiento de la SVM y 100 para test. Se utilizaron los siguientes 12 modelos de dispositivos móviles: IPhone 4s, IPhone 5s, Blackberry 8520, Huawei U8815, LG E400, LG P760, Nokia 800, Samsung GT-I9001, Samsung GT-I9100, Samsung GT-I8160P, Samsung GT-5830M y Sony C2105. Las imágenes cumplen las mismas restricciones que las cámaras del apartado anterior.

La Tabla 11.3 muestra los dispositivos móviles utilizados para los experimentos y la configuración de las cámaras.

Tabla 11.3: Configuración utilizada en las cámaras de los dispositivos móviles

Marca	Modelo	Resolución	Condiciones de captura
Apple	IPhone 4s (I1)	2 MP (2048×1536)	Tipo de escena: Cualquiera Orientación: Vertical Flash: Desabilitado Luz: Natural Balanceo de blancos: Auto Radio de zoom digital: 0 Tiempo de exposición: 0 seg Velocidad ISO: Automático
	IPhone 5s (I2)	3.15 MP (2048×1536)	
Black Berry	8520 (BB)	2 MP (1600×1200)	
Huawei	U8815 (HU)	2 MP (1600×1200)	
LG	E400 (LG1)	2 MP (1600×1200)	
	P760 (LG2)	3.15 MP (2048×1536)	
Nokia	800 (N1)	2 MP (1600×1200)	
Samsung	GT-I9001 (S1)	2 MP (1600×1200)	
	GT-I9100 (S2)	2 MP (1600×1200)	
	GT-I8160P (S3)	5 MP (2592×1944)	
	GT-5830M (S4)	5 MP (2592×1944)	
Sony Ericsson	C2105 (SE1)	2 MP (1600×1200)	

Los experimentos se han agrupado en 3 grupos con el objetivo de obtener conclusiones sobre el uso de los diferentes tipos de conjuntos de características, el tamaño del recorte, el número de dispositivos utilizados para la clasificación y la utilización de dispositivos del mismo fabricante. Los experimentos donde todos los dispositivos son del mismo fabricante ponen a prueba con exigencia las técnicas presentadas. Los componentes hardware y software de las cámaras de un mismo fabricante, en general, son muy parecidos o incluso iguales entre ellos, lo cual obviamente presenta graves dificultades o imposibilidad de distinción entre los distintos modelos de dispositivos móviles.

En la Tabla 11.4 se muestra el primer grupo de experimentos en el que se utilizan 7 modelos de dispositivos móviles de distinto fabricante.

Tabla 11.4: Identificación de la fuente para 7 dispositivos móviles

Características	Recorte	I1	HU	LG2	N1	BB	S1	SE1	% de Acierto
Todas	1024×768	93	96	80	94	91	70	85	86,54
Ruido	1024×768	41	42	35	18	40	40	62	37,67
Color	1024×768	24	37	20	40	31	19	44	29,27
IQM	1024×768	13	88	46	89	7	34	2	21,65
Wavelet Daubechies 8-tap	1024×768	95	96	96	94	92	76	93	91,46
Wavelet Haar	1024×768	95	87	97	70	86	56	91	81,84
Color + IQM + Wavelet Daubechies 8-tap	1024×768	93	94	90	90	90	53	85	83,67
Todas	800×600	91	96	84	92	95	56	85	84,41
Todas	640×480	90	95	84	89	88	51	88	82,15

Se prueban diferentes tipos de combinaciones de conjuntos de características. La mayoría de los experimentos se realizaron con un recorte de 1024×768, ya que como se mostró en los experimentos anteriores se estima un tamaño suficientemente grande para la obtención de buenos resultados. Asimismo, se consideraron los recortes de la parte central de las imágenes.

El experimento revela que las características de ruido, color e IQM por separado son inválidas, ya que a lo sumo se obtiene una tasa media de acierto del 37,67 % siendo inaceptable. Con las wavelets se realizaron dos experimentos utilizando distintos tipos de wavelet: (a) Daubechies 8-tap y (b) Haar. Los resultados muestran que Daubechies 8-tap obtiene mejores resultados que Haar y, a su vez, los mejores resultados (91,46 %).

Con respecto a las distintas combinaciones de características, se observa que al utilizar todas las características se obtienen buenos resultados (86,54 %), ya que aunque son ligeramente peores que el mejor resultado, la diferencia no es muy alta (un 4,92 %). Asimismo, puede verse que la tasa de acierto usando todas las características baja sutilmente cuanto menor es el tamaño del recorte.

La combinación de todas las características menos las de ruido obtiene una tasa media de acierto del 83,67 %. Estos resultados, no siendo malos, distan del obtenido con las wavelets y son peores que cuando se utiliza la combinación de todas las características.

Con objeto de seguir evaluando los resultados de la utilización de todos los conjuntos de características, en el siguiente conjunto de experimentos se utilizan 10 modelos de dispositivos móviles, algunos de ellos del mismo fabricante. Los resultados de los experimentos realizados pueden verse en la Tabla 11.5.

Tabla 11.5: Identificación de la fuente de adquisición para 10 dispositivos móviles

Características	**Tamaño de Recorte**	**I1**	**I2**	**HU**	**LG1**	**LG2**	**N1**	**BB**	**S1**	**S4**	**SE1**	**% de Acierto**
Daubechies 8-tap	1024×768	91	87	96	47	89	92	95	61	79	80	79,98
Haar	1024×768	84	76	81	43	78	64	64	41	82	89	68,04
Daubechies 8-tap	800×600	91	85	95	43	80	88	97	52	72	82	76,23
Daubechies 8-tap	640×480	92	77	99	42	84	90	90	45	73	84	74,86

Ya se comentó anteriormente que el hecho de que existan dispositivos del mismo fabricante y de similares características dificulta enormemente la tarea de clasificación, ya que las cámaras pueden ser iguales o prácticamente iguales. Como era de prever, se constata que a mayor número de dispositivos y siendo algunos del mismo fabricante, las tasas de acierto bajan para todos los casos (un 6,56 % en el mejor de los casos). Aun así, se considera que la bajada no es extremadamente pronunciada, teniendo en cuenta que hay 3 dispositivos más y sobre todo que hay 2 modelos del mismo fabricante. Es importante destacar que el dispositivo LG E400 tiene para todos los casos las tasas de acierto más bajas con gran diferencia con respecto al resto de dispositivos (un 43,71 % de tasa media de acierto). El 31 %, el 38 %, el 38 % y el 32 % de las imágenes del LG E400 (Optimus L3) fueron clasificadas como imágenes del LG P760 (Optimus L9) para

el primer, segundo, tercer y cuarto experimento de este conjunto, respectivamente. Esto indica claramente que hay un gran nivel de confusión entre las imágenes del LG E400 con respecto al LG P760, lo cual, entre otros casos, hace bajar notablemente la tasa de acierto general de cada experimento. De los resultados se desprende que la tecnología y componentes hardware y software de ambas cámaras pueden ser similares (aun habiendo dos modelos intermedios, el Optimus L5 y L7) o que el conjunto de características definido no permite discernir correctamente entre las dos cámaras. Asimismo, se constata, al igual que en el experimento anterior, que la familia de wavelets Haar no es adecuada para clasificar imágenes de dispositivos móviles.

Para indagar más profundamente sobre los resultados de distinción entre cámaras del mismo fabricante se realizó el siguiente conjunto de experimentos, en el que se utilizan 4 modelos de dispositivos móviles del fabricante Samsung. Los resultados de los experimentos realizados pueden verse en la Tabla 11.6.

Tabla 11.6: Identificación de la fuente para 4 dispositivos móviles del mismo fabricante

Características	**Tamaño de Recorte**	**S1**	**S2**	**S3**	**S4**	**% de Acierto**
Todas las características (Daubechies 8-tap)	1024×768	93	84	67	81	80,69
IQM + Color + Wavelet Daubechies 8-tap	1024×768	88	84	50	84	74,65
Todas las características (Daubechies 8-tap)	800×600	89	81	61	86	78,42
Todas las características (Daubechies 8-tap)	640×480	88	78	63	77	75,96

Primeramente se observa que los resultados utilizando todas las características menos las de ruido son peores que los obtenidos cuando se utilizan todas las características, al igual que se observó en los experimentos de la Tabla 11.4. Una vez dicho esto, se observa que los resultados obtenidos son bastante buenos, ya que en el mejor de los casos se obtiene un 80,69 % de tasa de acierto medio. Los resultados obtenidos para el dispositivo Samsung GT-S830 son más deficientes con respecto a los de los otros dispositivos, concretamente un 59,90 %, cuando la tasa de acierto en el peor de los casos del resto de dispositivos es del 81,71 %. El 20 %, el 33 %, el 24 % y el 23 % de las imágenes del Samsung GT-I8160P (Galaxy Ace 2) fueron clasificadas como imágenes del Samsung GT-S5830M (Galaxy Ace) para el primer, segundo, tercer y cuarto experimento de este conjunto, respectivamente. De hecho, este resultado concreto baja la mejor tasa de acierto obtenida en un 5,16 %. Análogamente al caso anterior, pueden suponerse las mismas conclusiones, aunque en este caso la similitud a todos los niveles de las cámaras tiene un mayor sentido, ya que no existen modelos intermedios en la serie Ace de Samsung entre los dos dispositivos, es decir, uno sucede al otro.

11.3 Síntesis del Capítulo

El objetivo principal de este capítulo ha sido presentar una técnica con el fin de identificar el tipo de dispositivo (escáner, computador o dispositivo móvil) o clase (marca y modelo) de la fuente de adquisición de la imagen, ésta última específicamente para dispositivos móviles. En primer lugar se han mostrado las características de Ruido, Color, IQM y Wavelets utilizados por la propuesta. Después se han descrito 36 experimentos clasificados en 5 series, con el fin de probar diferentes configuraciones de la técnica. En la configuración de los experimentos se ha tomado en cuenta, entre otras cosas, el futuro uso de la técnica por el analista forense en situaciones reales para crear experimentos con exigentes requerimientos prácticos. Finalmente, se han mostrado los principales resultados de la técnica de identificación propuesta.

Capítulo 12

Agrupamiento de Imágenes Digitales

En este capítulo se propone un algoritmo de agrupamiento. Como elementos para la clasificación se usa un conjunto de características obtenidas del ruido del sensor SPN. En términos generales, la principal diferencia con respecto a otras técnicas es que esta propuesta tiene en cuenta el proceso evolutivo de formación de grupos al calcular el coeficiente que determina la cohesión entre los elementos de un mismo grupo y la separación entre los diferentes grupos que se generan. Este capítulo está dividido en cuatro secciones. La Sección 12.1 presenta brevemente una visión general de los tipos de escenarios en los que se aplican las técnicas de identificación de la fuente de adquisición. El algoritmo propuesto se especifica en detalle en la Sección 12.2. Los experimentos con bancos de imágenes y sus resultados se presentan en la Sección 12.3. El capítulo finaliza con una breve síntesis de lo expuesto en el mismo.

12.1 Generalidades

El objetivo del análisis de grupos o agrupamiento de imágenes digitales es clasificar una colección de imágenes en conjuntos cuyas imágenes pertenecen a un mismo dispositivo llamados grupos sin información a priori.

El agrupamiento de imágenes puede llevarse a cabo mediante técnicas de aprendizaje supervisadas o sin supervisión. En el primer caso es indispensable conocer información del dispositivo a priori, es decir, se identifica claramente con la clasificación en escenarios cerrados en donde se requiere una fase de entrenamiento con las características extraídas de las imágenes y una segunda fase de clasificación conforme al resultado anterior. Sin embargo, en un caso real puede ser difícil contar con la cámara en cuestión o con un conjunto de fotografías tomadas por la misma para llevar a cabo un entrenamiento, de ahí la necesidad de técnicas de aprendizaje sin supervisión, que se corresponden directamente con los escenarios abiertos.

El agrupamiento tradicional está caracterizado por ser una técnica de aprendizaje sin supervisión. Sin embargo, existen algunos casos de agrupamiento supervisado en donde es posible aplicar una metodología anterior o posterior para mejorar el propio agrupamiento. De esta forma se evita que haya elementos de distintas clases en un mismo grupo, para lo cual es necesario contar con conocimiento previo del conjunto de datos. En [EZZ04] se trata este tema aunque cabe mencionar que esta investigación está enfocada en el uso de técnicas no supervisadas.

En la mayoría de los trabajos del estado del arte sobre agrupamiento de imágenes por métodos sin supervisión se considera al ruido del sensor SPN como el criterio más fiable para representar la huella digital de un dispositivo, es de ahí que utilizan concretamente el ruido PRNU como huella y la correlación normalizada como medida de similitud para lograr el agrupamiento de imágenes por dispositivo.

Para poder determinar la similitud entre objetos pertenecientes a un mismo grupo existen medidas de distancia como pueden ser: distancia euclidiana, distancia Manhattan y distancia Chebychev, entre otras. Alternativamente, es posible usar funciones de similitud $S(X_i, X_j)$, las cuales comparan dos vectores X_i y X_j en forma simétrica, es decir, $S(X_i, X_j) = S(X_j, X_i)$. Estas funciones alcanzan valores altos cuando X_i y X_j son similares. Una de las medidas más utilizadas en la identificación de fuente de imágenes es la correlación normalizada [Blo08] [Fri09] [Li10b] [CAPI10] definida como:

$$corr\,(X_i, X_j) = \frac{X_i - \overline{X_i} \;\odot\; X_j - \overline{X_j}}{X_i - \overline{X_i} \;\cdot\; X_j - \overline{X_j}} \tag{12.1}$$

donde $\overline{X_i}$ y $\overline{X_j}$ representan la media del vector, $X_i \odot X_j$ es el producto punto de dos vectores y $\|X_i\|$ es la norma L_2 de X_i. Dado que el patrón del ruido del sensor es una matriz bidimensional, previamente a la aplicación de las funciones del cálculo de la correlación, se realiza una transformación a vector unidimensional.

12.2 Especificación del Algoritmo

El algoritmo de agrupamiento sin supervisión está basado en el propuesto en [CAPI10]. Se trata de una combinación entre un agrupamiento jerárquico y un agrupamiento plano. Es decir, a pesar de formar una estructura de dendograma con cada iteración del algoritmo, al final los grupos son tomados como entidades sin relación alguna ya que cada uno de ellos debe corresponder a un dispositivo específico.

La estructura general del algoritmo de agrupamiento propuesto se muestra en la Figura 12.1, siendo N el número de imágenes y q el número de iteración (comienza en 0).

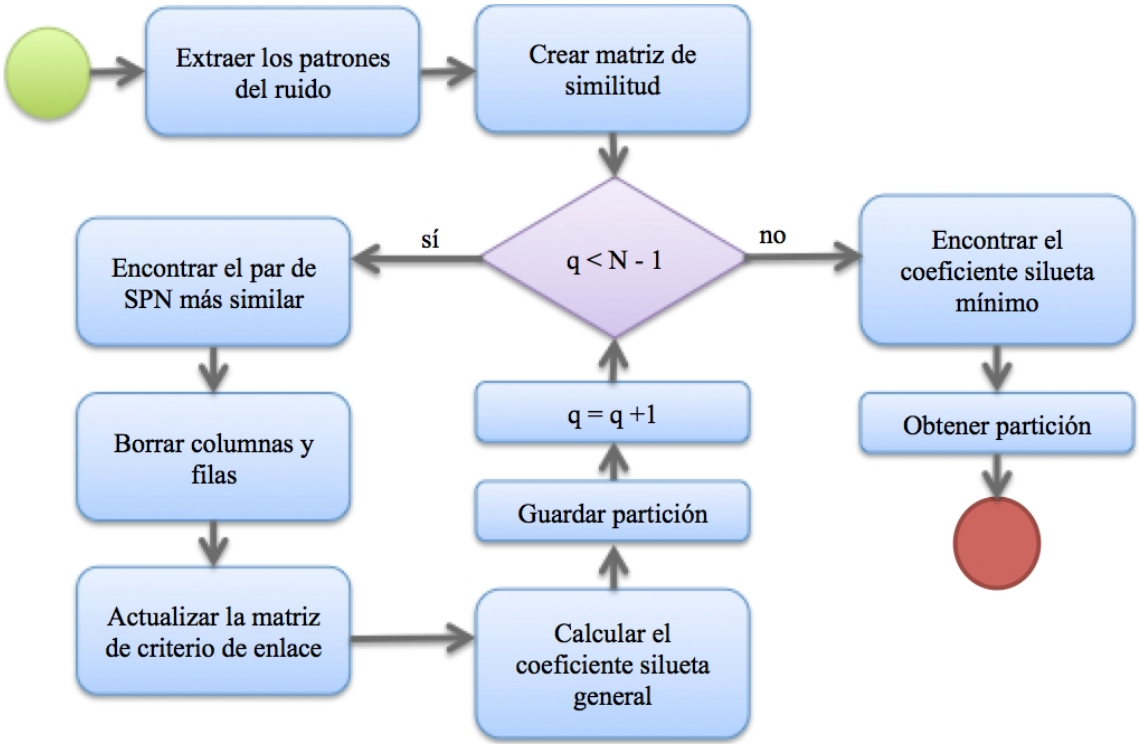


Figura 12.1: Algoritmo de agrupamiento

Previo a realizar el agrupamiento, es necesario obtener los patrones de ruido del sensor del conjunto de imágenes I como se propone en [AG15], utilizando el algoritmo de extracción y el parámetro de supresión de ruido $s_0 = 5$ que controla que tan fuerte es:

$$n^{(i)} = I^{(i)} - F \quad I^{(i)} \tag{12.2}$$

donde $i = 1, \ldots, N$, $n^{(i)}$ es el patrón del ruido de cada imagen i, $I^{(i)}$ es la imagen con ruido i y F es el filtro de extracción del ruido basado en la Transformada Wavelet. La extracción del SPN se basa en el algoritmo desarrollado en el Capítulo 13. En la propuesta no se ha utilizado ningún algoritmo de mejoramiento de ruido, como los propuestos por [CAPI10] y [Li10b]. El filtro de Wiener en el dominio de la frecuencia es suficiente para eliminar la mayoría de los detalles de la escena presentes al extraer el SPN.

Para cada uno de los N ruidos $(n_1, ..., n_N)$ se obtiene el valor de correlación usando la Ecuación 12.1 y esto genera una matriz de similitud H de $N \times N$. Dicha matriz es simétrica y está compuesta de unos en su diagonal principal (ya que la correlación de un ruido consigo mismo es 1) . Una vez generada la matriz no es necesario volver a calcular las correlaciones entre ruidos a lo largo del algoritmo de agrupamiento ahorrando tiempo y capacidad de procesamiento.

El algoritmo de agrupamiento jerárquico seleccionado encuentra dentro de la matriz H el par de ruidos k y l con el valor de correlación más alto. Cabe mencionar que los valores de correlación en la diagonal principal no son tomados en cuenta. A continuación, las filas y columnas correspondiente a k y l son eliminadas y tanto una nueva fila como

una nueva columna son agregadas a la matriz. Los valores de esta nueva fila y columna son el resultado de una función de criterio de enlace. La función elegida para este trabajo fue el criterio de enlace promedio, puesto que mejora los resultados de otros criterios de enlace como el criterio simple o el criterio completo, tal y como se sugiere en [CAPI10]. La Ecuación 12.3 muestra la función del criterio de enlace promedio entre dos grupos A y B:

$$H(A,B) = \frac{1}{\|A\| \, \|B\|} \sum_{ni \in A, nj \in B} corr(n_i, n_j) \tag{12.3}$$

donde el valor $corr(n_i, n_j)$ es calculado con la Ecuación 12.1 y puede ser tomado de la matriz H para simplificar el procesamiento computacional, $\|A\|$ es la cardinalidad del grupo A y $\|B\|$ es la cardinalidad del grupo B.

Cada iteración del algoritmo toma los dos grupos con el valor de correlación más alta en la matriz y mezcla los objetos contenidos en éstos para crear un nuevo grupo, al mismo tiempo que almacena el estado de los distintos grupos en particiones $P_0, ..., P_{N-1}$ con el objetivo de conocer el contenido de los grupos en cada momento. En el agrupamiento jerárquico, el resultado final del algoritmo es un grupo que contiene a todos los objetos. Sin embargo, en este trabajo cada grupo debería representar un dispositivo al final de la ejecución. Por este motivo, se usa el coeficiente silueta como medida de validación de grupos. El coeficiente silueta mide el índice de similitud entre los elementos de un mismo grupo (cohesión) y la similitud entre los elementos de un grupo con respecto a los demás (separación). A diferencia de Caldelli et al. [CAPI10], en la propuesta el cálculo del coeficiente silueta se realiza por cada grupo contenido en la partición P_i y no por cada patrón del ruido, como se observa en la Ecuación 12.4.

$$s_j = \text{máx}(b_j) - a_j \tag{12.4}$$

donde a_j (cohesión) es la correlación promedio entre todos los patrones de ruido dentro del grupo c_j y b_j (separación) es la correlación promedio de los patrones de ruido contenidos en el grupo c_j con respecto a los patrones de ruidos en los grupos restantes. Se toma el grupo vecino más cercano, es decir, aquel con la correlación más alta.

Como puede verse la forma de cálculo del coeficiente silueta varía notablemente con respecto a la propuesta de [CAPI10]. En la propuesta para cada iteración del algoritmo se calculan tantos coeficientes siluetas como grupos existan formados en esa iteración y no tantos coeficientes silueta como imágenes totales haya. Según evoluciona el algoritmo y los grupos van aumentando su número de imágenes, se calcula un número menor de coeficientes siluetas. Además, según se van formando los grupos en cada iteración del algoritmo, cada grupo se toma como una entidad única para el cálculo del coeficiente silueta. Por tanto, no se tiene en cuenta cada uno de los ruidos independientes que forman cada grupo, ya que lo que se desea medir es la cohesión y separación entre grupos y no entre imágenes independientes. Una vez se tiene la idea de grupo como entidad unitaria para el cálculo del coeficiente silueta, se calcula para grupo teniendo en cuenta la obtención de la máxima separación con respecto a otros grupos y una alta cohesión de todos los elementos de cada uno de los grupos formados, como puede observarse en la Ecuación 12.4

Para cada iteración q del algoritmo se obtiene una medida global de todos los coeficientes siluetas calculados a partir de los K grupos; esto equivale a promediar los valores s_j en q. La Ecuación 12.5 muestra dicho cálculo.

$$SC_q = \frac{1}{K} \sum_{j=1}^{K} s_j \tag{12.5}$$

Una vez concluido el agrupamiento jerárquico se procede a buscar el SC_q con el valor mínimo, lo cual indica que los grupos de la partición P_q^* están en un nivel de correlación mayor. El número de grupos en ese instante debería corresponder al número real de dispositivos. El objetivo de almacenar la partición en cada momento del algoritmo es evitar volver a ejecutar el agrupamiento puesto que se tiene información de todos los grupos en cada iteración q.

En el Algoritmo 11 se muestra el pseudocódigo de la propuesta.

Algoritmo 11: Algoritmo de agrupamiento

1. Calcular el patrón del ruido $n^{(i)}$ de cada imagen donde $i \in 1, ..., N$;
2. Generar matriz de similitud $H \in R^{NxN}$;
3. **foreach** $q \in 1, ..., N-1$ **do**
4. Encontrar el par de grupos $H(k, l)$ con la mayor similitud;
5. Eliminar el par de filas y columnas correspondientes a los grupos k y l;
6. Calcular los valores del nuevo grupo usando el criterio de enlace promedio y agregar tanto la fila como columna correspondientes;
7. Determinar el coeficiente silueta global SC_q;
8. Almacenar la partición P_q;
9. Encontrar la partición donde el coeficiente silueta mínimo $min_q(SC_q)$;

Como se mencionó anteriormente, el objetivo del agrupamiento es agrupar objetos en un entorno no supervisado (escenario cerrado). Sin embargo, en la metodología elegida es factible llevar a cabo una etapa de entrenamiento y una de clasificación para reducir la complejidad computacional y, por ende, reducir el tiempo de ejecución del algoritmo. Para ello es necesario partir el conjunto de imágenes en dos subconjuntos: uno de entrenamiento I_e y otro de clasificación I_c. El subconjunto de entrenamiento es procesado por el algoritmo descrito anteriormente para obtener al final K grupos que representan a los dispositivos. Posteriormente, se calcula un centroide c_j por cada grupo promediando los m patrones de ruido contenidos en éste. A continuación, se obtiene el valor de correlación entre cada SPN del subconjunto I_c y cada centroide c_j. La imagen es clasificada entonces en el grupo donde haya existido un valor mayor de correlación. La clasificación propuesta podemos observarla en el Algoritmo 12.

Algoritmo 12: Algoritmo de agrupamiento con etapa de entrenamiento

1. Calcular el centroide c_j de cada grupo donde $j \in 1, ..., K$ y $c_j = \frac{1}{m}\sum_1^m n$;
2. Calcular el patrón del ruido n_i del subconjunto de clasificación $I_c \subset I$;
3. **foreach** $n_i \in i_c$ **do**
4. Clasificar n_i en el grupo de mayor correlación
5. $f_j = \arg\max_j corr(n_i, c_j)$;

12.3 Experimentos y Resultados

Los experimentos fueron realizados con un conjunto total de 1350 fotografías de 9 diferentes modelos de cámaras de dispositivos móviles (150 fotografías de cada modelo) 7 dispositivos son de diferentes fabricantes (Apple iPhone 5, Huawei U8815, LG E400, Samsung GT-S5830M, Zopo ZP980, Sony ST25a y Nokia 800 Lumia) y los 2 dispositivos restantes fueron fabricados por Sony (Sony ST25i y Sony C2105).

Todas las imágenes fueron recortadas a 1024×1024 píxeles debido a que las imágenes tienen diferentes dimensiones y trabajar con éstas a tamaño completo tiene un mayor coste computacional.

Para disminuir el porcentaje de errores en el agrupamiento todas las imágenes poseen una orientación horizontal siendo necesario aplicar una rotación de 90° a las imágenes capturadas en posición vertical. Las escenas de las fotografías fueron elegidas de forma aleatoria y son tanto de interiores como de exteriores. Asimismo, fueron tomadas en diferentes momentos y lugares con el objetivo de simular un escenario más realista.

En la extracción del patrón del ruido de todas las imágenes se utilizó el promedio a cero de filas y columnas y los 3 canales de color *Red-Green-Blue* (RGB) fueron convertidos a una sola matriz en escala de grises. Adicionalmente, todos los experimentos se llevaron a cabo utilizando el filtro de Wiener en el dominio de la frecuencia. La Figura 12.2 muestra un diagrama del preprocesamiento realizado a la imagen.

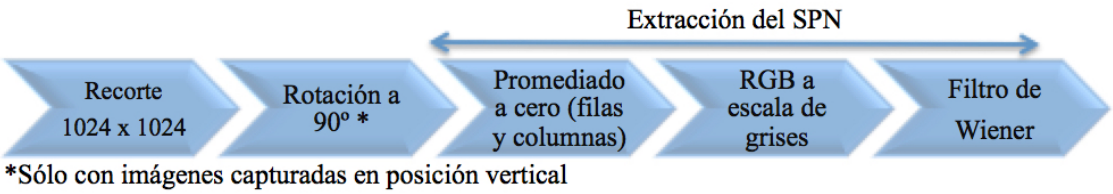


Figura 12.2: Preprocesamiento de una imagen

Para medir el grado de certeza en los resultados se utilizó la tasa de verdaderos positivos TPR. El TPR promedio para cada uno de los siguientes experimentos se calcula, computando para cada grupo el número de fotos que han sido bien clasificadas (TPR de cada grupo) y promediando los TPRs de todos los grupos resultantes (si hay menos grupos que dispositivos se promedia teniendo en cuenta el número de dispositivos). Para

calcular el TPR de cada grupo, hay que ver en el grupo el dispositivo que tiene el mayor número de imágenes con respecto al total de imágenes por dispositivo, siendo ese el grupo predominante del dispositivo. Posteriormente, se calcula el porcentaje de fotos que han sido bien clasificadas para ese dispositivo en ese grupo. Realmente en la inmensa mayoría de los casos puede verse fácilmente que un grupo se asocia a uno o varios dispositivos como puede apreciarse en las matrices de las Tablas 12.1, 12.2 y 12.3. Si hay varios grupos con el mismo número de fotos de un dispositivo o un grupo con igual número de fotos de varios dispositivos y, a su vez, siendo éstos los máximos, se toma como grupo predominante para el dispositivo el que se desee de entre las distintas opciones. Puede darse el caso de que si hay un grupo de más, un grupo no sea predominante de ningún dispositivo (Tabla 12.2) y su TPR para ese grupo sea 0. También puede darse el caso de que haya un grupo de menos (Tabla 12.3), en cuyo caso se tiene en cuenta la asociación del grupo a un dispositivo y se utiliza para el promedio el número de dispositivos, como se indicó anteriormente. En las Tablas 12.1, 12.2 y 12.3 pueden verse ejemplos que ilustran el cálculo del TPR para los tres casos que pueden darse.

Tabla 12.1: TPR con igual número de dispositivos que grupos

Marca - Modelo	**Grupo 1**	**Grupo 2**	**Grupo 3**	**Grupo 4**	**Grupo 5**	**TPR Promedio**
Apple - Iphone 5	49	0	0	1	0	
Huawei - U8815	0	50	0	0	0	
LG - E400	0	1	49	0	0	
Nokia - 800 Lumia	0	0	0	50	0	
Samsung - GT5830m	0	0	0	0	50	
TPR por grupo	**98 %**	**100 %**	**98 %**	**100 %**	**100 %**	**99,2 %**

Tabla 12.2: TPR con menor número de dispositivos que grupos

Marca - Modelo	**Grupo 1**	**Grupo 2**	**Grupo 3**	**Grupo 4**	**TPR promedio**
Apple - Iphone 5	100	0	0	0	
Huawei - U8815	0	100	0	0	
LG - E400	0	0	97	3	
TPR por grupo	**100 %**	**100 %**	**97 %**	**0 %**	**99 %**

Tabla 12.3: TPR con mayor número de dispositivos que grupos

Marca - Modelo	**Grupo 1**	**Grupo 2**	**Grupo 3**	**Grupo 4**	**TPR Promedio**
Apple - Iphone 5	100	0	0	0	
Huawei - U8815	0	100	0	0	
LG - E400	0	0	100	0	
Nokia - 800 Lumia	100	0	0	0	
Samsung - GT 5830M	0	0	0	100	
TPR por grupo	**100 %**	**100 %**	**100 %**	**100 %**	**80 %**

En los resultados de los experimentos se consideran 3 posibles casos: a) Número de grupos identificados igual al número de dispositivos, b) número de grupos identificados mayor que el número de dispositivos, y c) número de grupos identificados menor que el número de dispositivos. Aunque el primer caso es el ideal, en el segundo caso se pueden obtener clasificaciones en las que no se mezclen distintos tipos de dispositivos en un mismo grupo.

Los experimentos se dividen según los siguientes criterios:

- Comparativa entre tomar la región de 1024×1024 desde la esquina o desde el centro de la fotografía.
- Distribución simétrica o asimétrica de las fotografías (mismo o diferente número de fotografías por dispositivo).
- Comparativa de agrupamiento entre dispositivos del mismo fabricante pero distinto modelo.
- Entrenar y después llevar a cabo una clasificación.

12.3.1 Evaluación de la Región de Recorte de la Imagen

Se realizaron varios experimentos para comparar los resultados entre recortar la imagen desde el centro o desde la esquina superior izquierda, teniendo este último criterio un TPR mayor (salvo para el caso de 7 dispositivos con 100 imágenes cuya diferencia es mínima). La Tabla 12.4 muestra el TPR en función del número distinto de dispositivos utilizados y el número de fotos utilizadas por dispositivo. Todos los dispositivos tienen el mismo número de fotos. En la Tabla 12.4 se puede observar como el TPR aumenta en el caso del recorte en el centro a medida que se agrupan más dispositivos, mientras que para el recorte por la esquina se mantienen los buenos resultados para el caso de 50 imágenes por dispositivo y los resultados son dispares para el caso de 100 imágenes por dispositivo. Aunque [Li10a] menciona que las áreas de las esquinas son más propensas a estar saturadas y, por tanto, puede verse afectado el patrón del ruido, el algoritmo propuesto demuestra lo contrario en el agrupamiento de las imágenes.

Tabla 12.4: TPR para agrupamiento simétrico en función del número distinto de dispositivos y del número de fotos por dispositivo

Número	**Recorte desde Esquina Superior**			**Recorte desde el Centro**		
de Fotos	**3**	**5**	**7**	**3**	**5**	**7**
50	99,33 %	99,20 %	99,71 %	66,67 %	80 %	99,71 %
100	74,25 %	100 %	87,13 %	66,67 %	80 %	87,13 %

12.3.2 Agrupamiento Simétrico

A continuación se va a mostrar información adicional sobre tres de los experimentos de agrupamiento simétrico realizados anteriormente cuyos resultados generan igual, menor y mayor número de grupos que dispositivos hay respectivamente para su clasificación. Para cada experimento, una vez aplicado el algoritmo de agrupamiento propuesto en este trabajo, se muestra una gráfica para cada grupo generado. Esta gráfica muestra el grado de correlación del patrón del ruido de todas las imágenes utilizadas en cada experimento con respecto al centroide (promedio de todos los patrones de ruido contenidos en el grupo) de cada grupo generado por el algoritmo.

El primer experimento realiza un agrupamiento simétrico de 3 dispositivos, 50 imágenes y recortadas desde la esquina superior izquierda. En la Tabla 12.5 se muestra la matriz de confusión donde se detallan los grupos generados y las imágenes incluidas en cada uno de ellos. Como se aprecia en la Tabla 12.5 su TPR es del 99,33 %.

Tabla 12.5: Matriz de confusión del agrupamiento simétrico de 3 dispositivos

Marca - Modelo	Grupo 1	Grupo 2	Grupo 3	TPR Promedio
Apple Iphone 5	50	0	0	
Huawei U8815	0	50	0	
LG E400	0	1	49	
TPR por grupo	**100 %**	**100 %**	**98 %**	**99,33 %**

Como puede observarse se generan igual número de grupos que de dispositivos, lo cual en principio es indicio de posible obtención de un buen resultado. La clasificación para este caso es casi perfecta, salvo con la excepción de que en el grupo 2 hay una imagen del LG E400 que debería haberse clasificado en el grupo 3. Las Figuras 12.3 a 12.5 muestran las 3 gráficas de correlación descritas anteriormente para cada uno de los grupos generados.

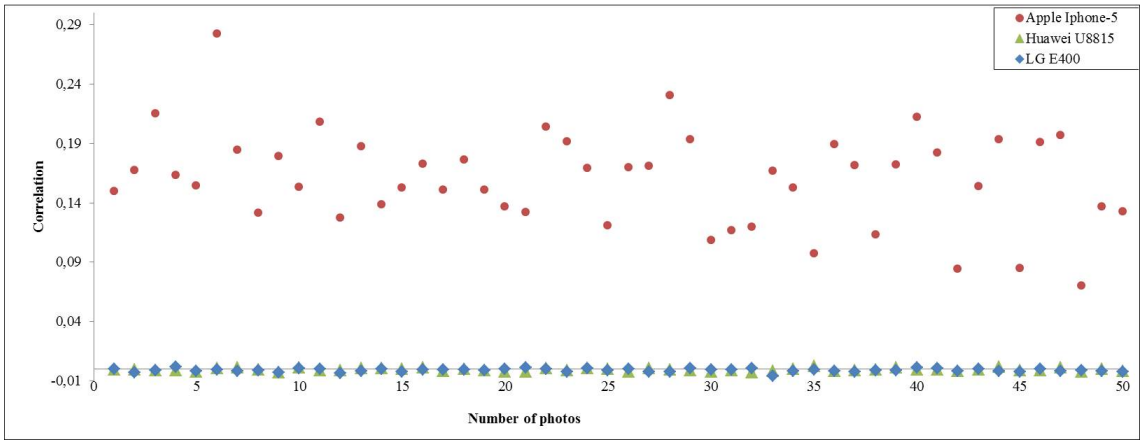


Figura 12.3: Correlación de imágenes con respecto al centroide del grupo 1

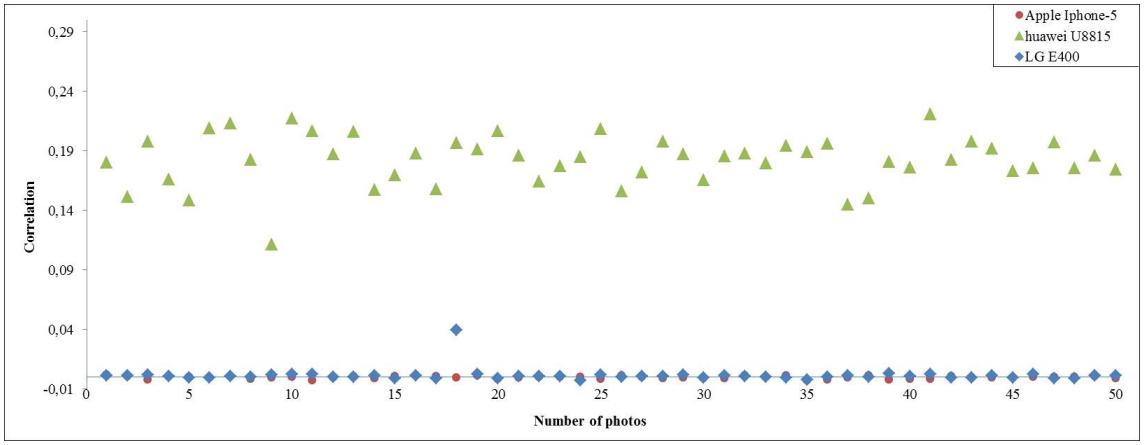


Figura 12.4: Correlación de imágenes con respecto al centroide del grupo 2

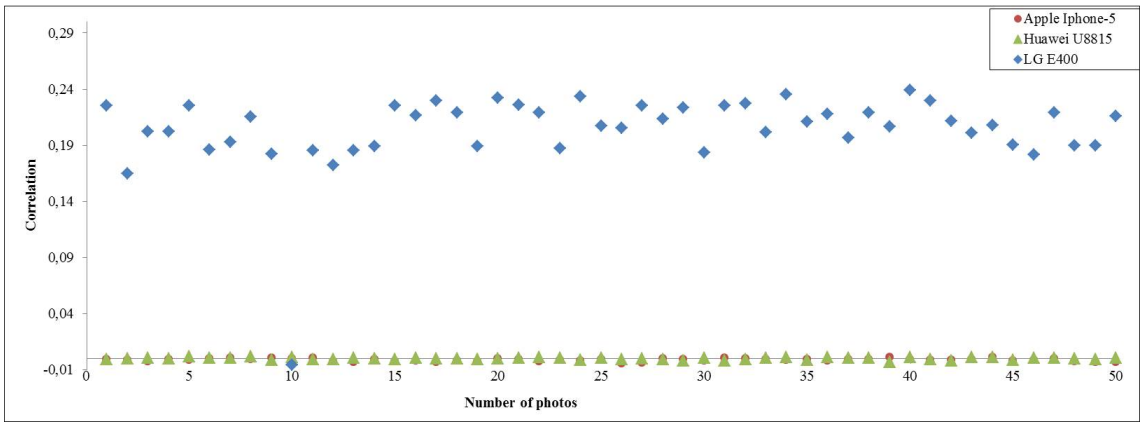


Figura 12.5: Correlación de imágenes con respecto al centroide del grupo 3

En la Figura 12.3 puede apreciarse que las correlaciones de un dispositivo con respecto a las restantes son distantes. Por tanto, genera correctamente un grupo con imágenes de un único dispositivo. En la Figura 12.4 puede observarse como hay una imagen del LG E400 cuyo grado de correlación no está en consonancia con las restantes del mismo dispositivo y las del Huawei U8815. Esta imagen concreta es la que se clasificó erróneamente, obteniendo un grado de correlación con respecto al centroide menor que las imágenes restantes de su propio grupo, pero distinto de cero y notablemente mayor (del orden de 20 a 400 veces aproximadamente) a la correlación de las imágenes restantes. En la Figura 12.5 puede observarse como todas las imágenes del LG E400 tienen una correlación similar salvo una que prácticamente tiene un valor de correlación cero con respecto al centroide del grupo 3 y se corresponde con la imagen mal clasificada. En la Figura 12.3 puede apreciarse que las correlaciones de un dispositivo con respecto a las restantes son distantes. Por tanto, genera correctamente un grupo con imágenes de un único dispositivo. En la Figura 12.4 puede observarse como hay una imagen del LG E400 cuyo su grado de correlación no está en consonancia con las restantes del mismo dispositivo y las del Huawei U8815. Esta

imagen concreta es la que se clasificó erróneamente, obteniendo un grado de correlación con respecto al centroide menor que las imágenes restantes de su propio grupo, pero distinto de cero y notablemente mayor (del orden de 20 a 400 veces aproximadamente) a la correlación de las imágenes restantes. En la Figura 12.5 puede observarse como todas las imágenes del LG E400 tienen una correlación similar salvo una que prácticamente tiene un valor de correlación cero con respecto al centroide del grupo 3 y se corresponde con la imagen mal clasificada.

El segundo experimento realiza un agrupamiento simétrico de 5 dispositivos, 50 imágenes y recorte central. Como se aprecia en la Tabla 12.6 su TPR es del 80 %. En la Tabla 12.6 se muestra una matriz de confusión donde se detallan los grupos generados y las imágenes incluidas en cada uno de ellos.

Tabla 12.6: Matriz de confusión del agrupamiento simétrico de 5 dispositivos

Marca - Modelo	Grupo 1	Grupo 2	Grupo 3	Grupo 4	TPR Promedio
Apple - Iphone 5	50	0	0	0	
Huawei - U8815	0	50	0	0	
LG - E400	0	0	50	0	
Nokia - 800 Lumia	50	0	0	0	
Samsung - GT 5830m	0	0	0	50	
TPR por grupo	**100 %**	**100 %**	**100 %**	**100 %**	**80 %**

Como puede observarse se genera un número de grupos menor al de dispositivos, lo cual implica que al menos uno de los grupos no es puro, es decir, tiene imágenes de al menos dos dispositivos. La clasificación para este caso es totalmente correcta para tres de los cuatro grupos generados. En cambio, en el grupo 1 están todas las imágenes de los dispositivos Apple iPhone 5 y Nokia 800 Lumia. Las Figuras 12.6 a 12.9 muestran las 4 gráficas de correlación descritas anteriormente para cada uno de los grupos generados.

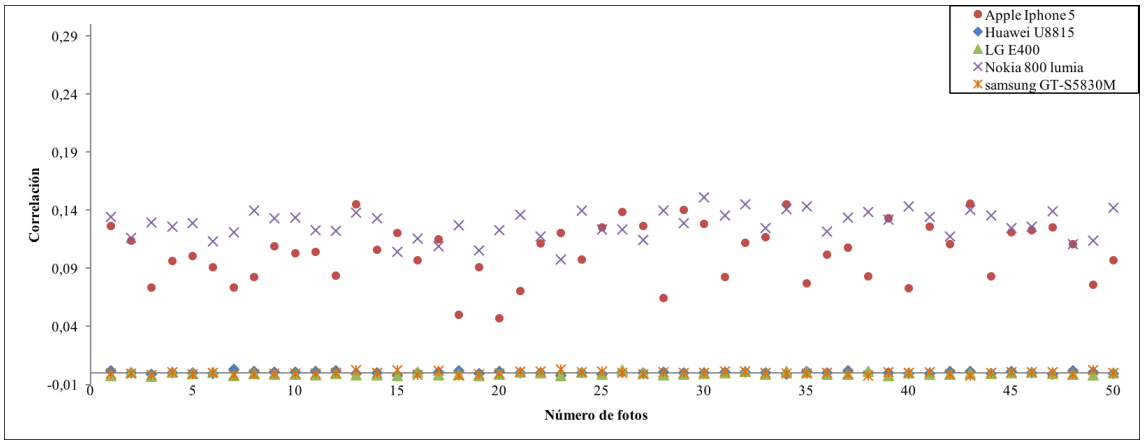


Figura 12.6: Correlación de imágenes con respecto al centroide del grupo 1

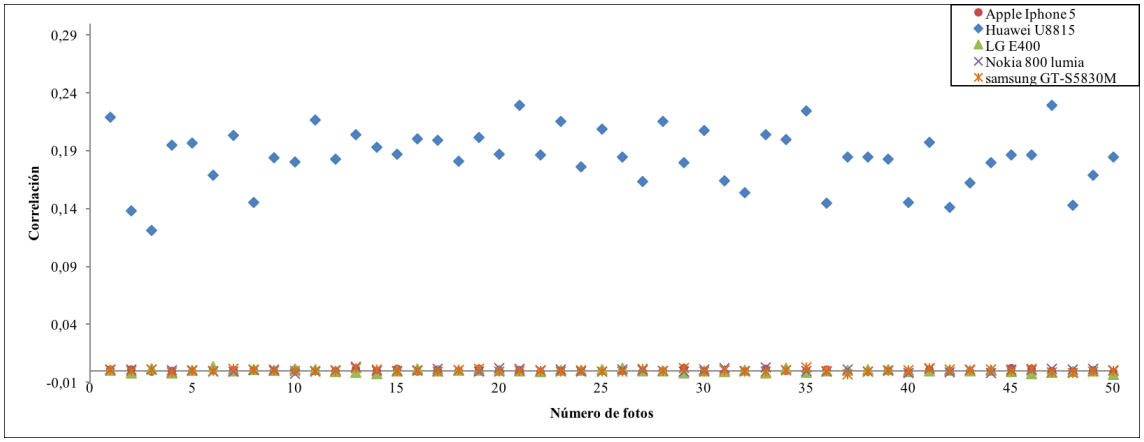


Figura 12.7: Correlación de imágenes con respecto al centroide del grupo 2

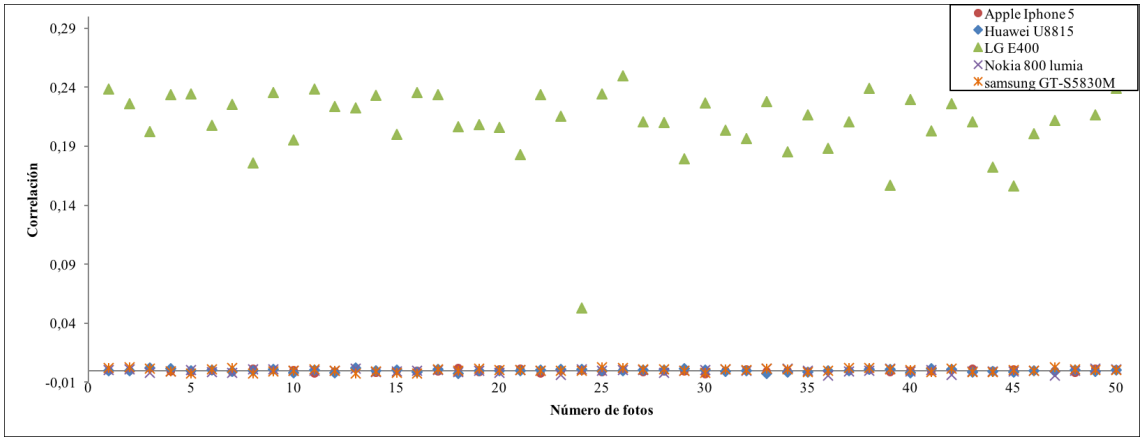


Figura 12.8: Correlación de imágenes con respecto al centroide del grupo 3

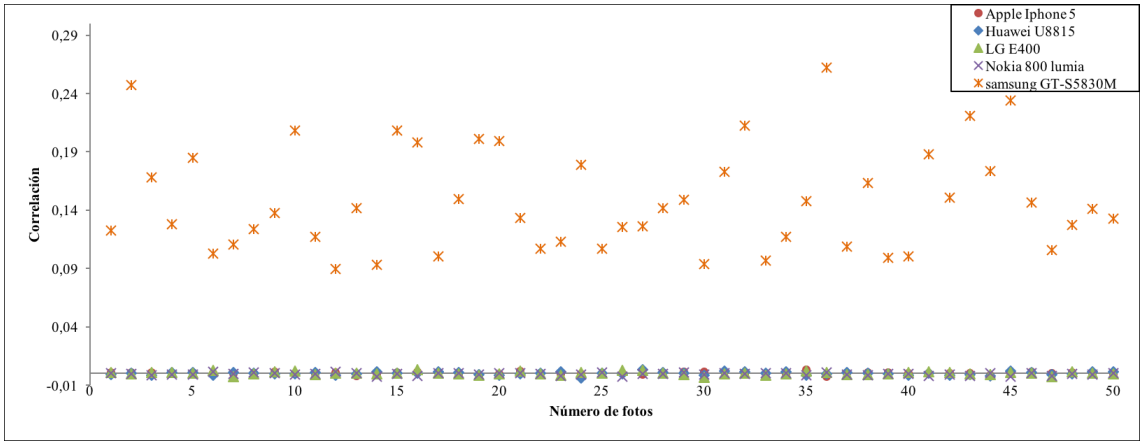


Figura 12.9: Correlación de imágenes con respecto al centroide del grupo 4

Como puede apreciarse en las Figuras 12.7 a 12.9, que corresponden a grupos con todas las imágenes de un único dispositivo, la correlación de las imágenes del dispositivo clasificado correctamente con respecto a los demás es distante. Para estos casos la correlación con respecto al centroide de las imágenes fuera del grupo es aproximadamente cero en todos los casos. En la Figura 12.6 puede observarse como la correlación de las imágenes del Apple iPhone 5 Huawei U8815 son similares formando el grupo 1 y existe gran diferencia con la correlación del resto de imágenes, que es cercana a cero.

El tercer experimento realiza un agrupamiento simétrico de 7 dispositivos, 100 imágenes y recorte desde el borde. Como se aprecia en la Tabla 12.7 su TPR es del 87,13 %. En la Tabla 12.7 se muestra una matriz de confusión donde se detallan los grupos generados y las imágenes incluidas en cada uno de ellos.

Tabla 12.7: Matriz de confusión del agrupamiento simétrico de 7 dispositivos

Marca - Modelo	**Grupos**								TPR
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**Promedio**
Apple - Iphone 5	100	0	0	0	0	0	0	0	
Huawei - U8815	0	100	0	0	0	0	0	0	
LG - E400	0	0	97	0	0	0	0	3	
Nokia - 800 Lumia	0	0	0	100	0	0	0	0	
Samsung - GT 5830m	0	0	0	0	100	0	0	0	
Sony - ST25A	0	0	0	0	0	100	0	0	
Zopo - ZP980	0	0	0	0	0	0	100	0	
TPR por grupo	**100 %**	**100 %**	**97 %**	**100 %**	**100 %**	**100 %**	**100 %**	**0 %**	**87,13 %**

Como puede observarse se genera un número de grupos mayor que el de dispositivos. La clasificación para este caso es totalmente correcta para seis de los ocho grupos generados. Sin embargo, los grupos 3 y 8 contienen imágenes del LG E400. A pesar de que las imágenes del LG E400 se dividieron en dos grupos, éstos sólo poseen imágenes de un único dispositivo, lo cual es un aspecto positivo a tener en cuenta. La Figuras 12.10 a 12.17 muestran las 8 gráficas de correlación descritas anteriormente para cada uno de los grupos generados.

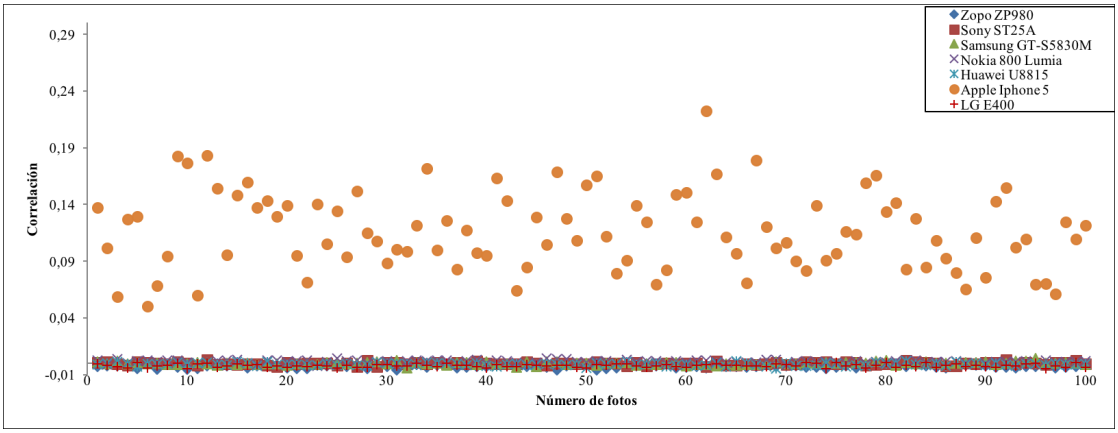


Figura 12.10: Correlación de imágenes con respecto al centroide del grupo 1

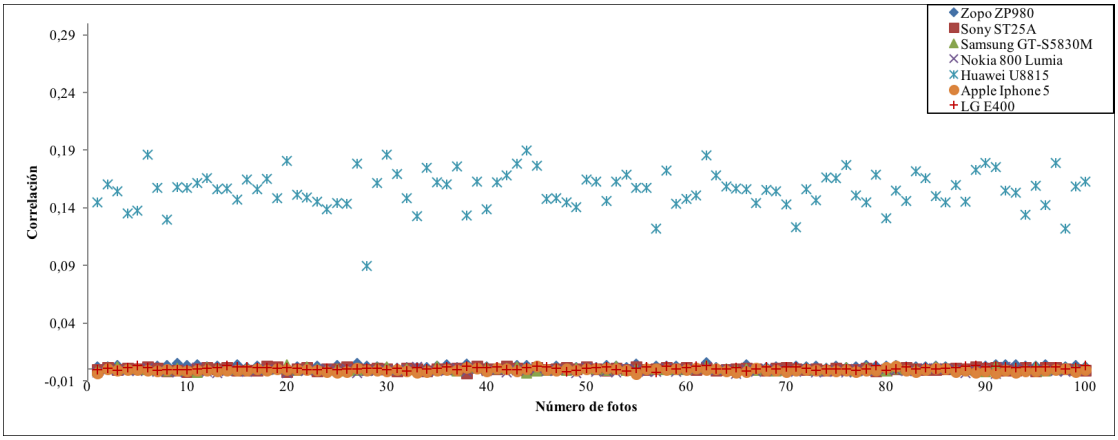


Figura 12.11: Correlación de imágenes con respecto al centroide del grupo 2

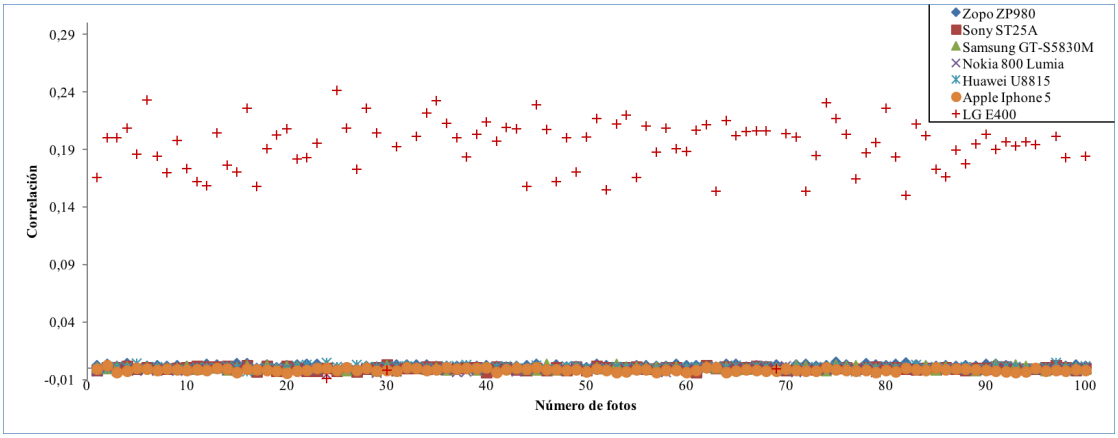


Figura 12.12: Correlación de imágenes con respecto al centroide del grupo 3

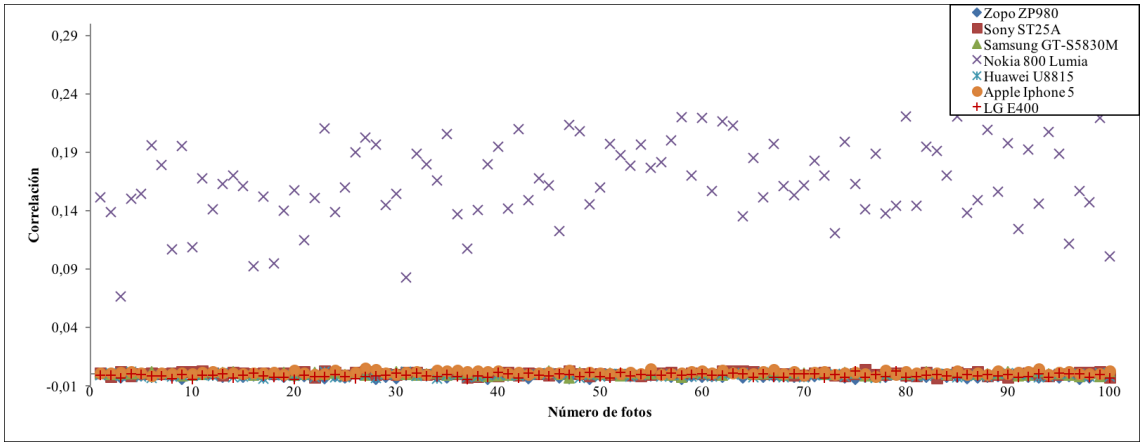


Figura 12.13: Correlación de imágenes con respecto al centroide del grupo 4

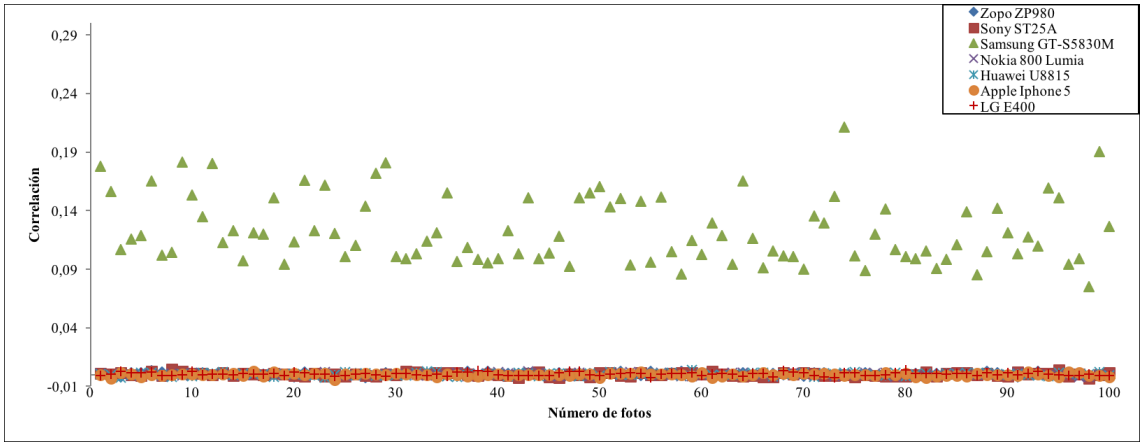


Figura 12.14: Correlación de imágenes con respecto al centroide del grupo 5

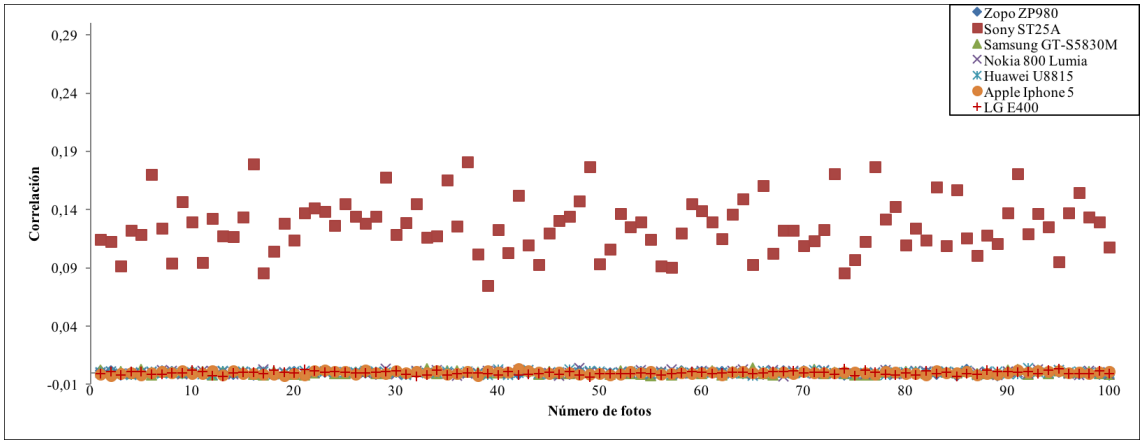


Figura 12.15: Correlación de imágenes con respecto al centroide del grupo 7

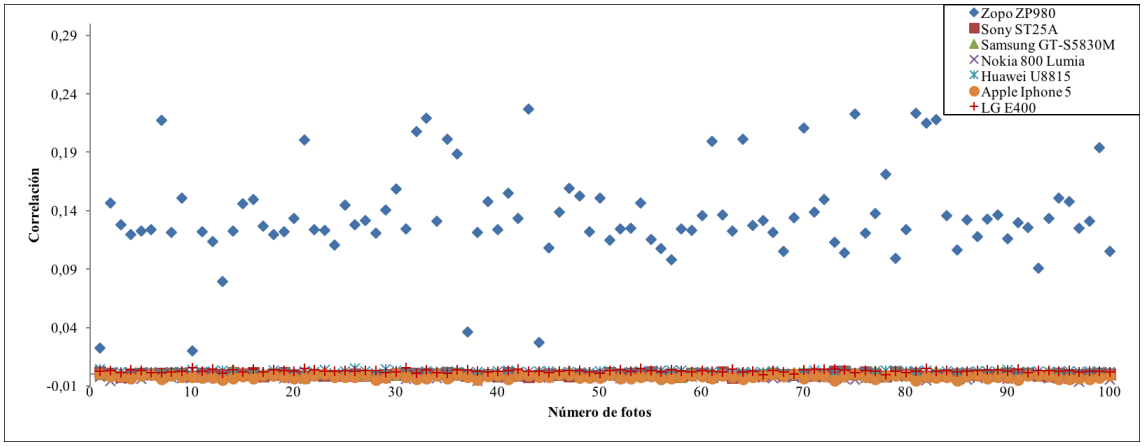


Figura 12.16: Correlación de imágenes con respecto al centroide del grupo 8

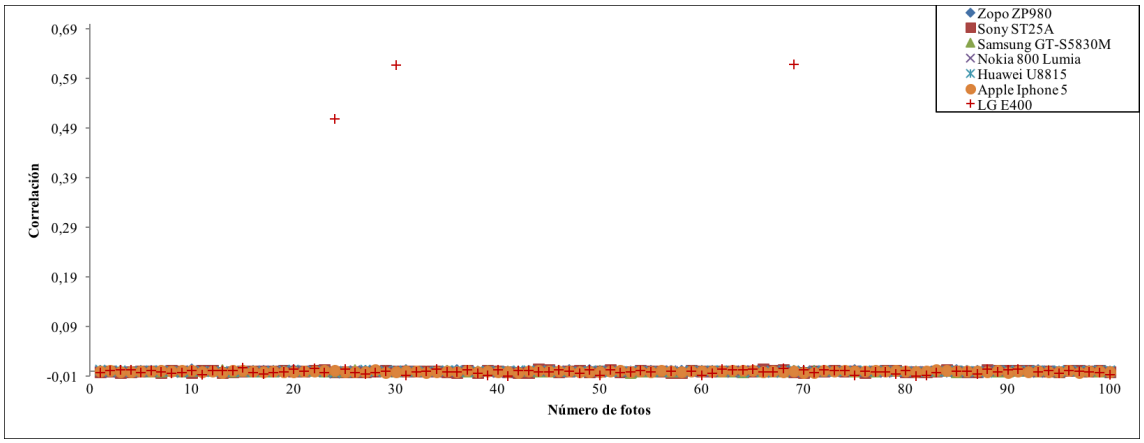


Figura 12.17: Correlación de imágenes con respecto al centroide del grupo 9

Como puede observarse en las Figuras 12.10 a 12.16, que se corresponden a grupos que se generaron correctamente, la correlación del dispositivo clasificado con respecto a los demás es distante. Para estos casos la correlación con respecto al centroide de las imágenes fuera del grupo se aproxima a cero, al igual que en experimentos anteriores.

En la Figura 12.12 hay 97 imágenes del LG E400 con una correlación notablemente superior a la del resto de imágenes. Existen 3 imágenes del LG E440 cuya correlación es prácticamente cero y muy distante del resto de las fotografías del mismo dispositivo en este grupo, conformando éstas el grupo 8.

En la Figura 12.17, puede apreciarse el resultado del agrupamiento de estas tres imágenes en un grupo independiente. La correlación de estas tres imágenes es notablemente superior a la del resto de imágenes, que tienen correlaciones próximas a cero.

En un escenario cerrado no es muy probable contar con el mismo número de imágenes de cada dispositivo a identificar. Por esa razón se realizaron experimentos en donde los conjuntos de imágenes por cada dispositivo no poseen una distribución simétrica para

comprobar la adaptabilidad del algoritmo propuesto en un escenario real.

En las Tablas 12.8 y 12.9 se presentan los resultados obtenidos al agrupar las imágenes de 5 y 7 dispositivos, respectivamente. El número de imágenes por dispositivo es variado y aún así podemos observar un alto grado de acierto (97,28 % TPR promedio de los experimentos de las Tablas 12.8 y 12.9). Como se puede observar en los casos de número de imágenes asimétrico se ha experimentado con grupos de bastante disparidad numérica y en algunos casos con grupos pequeños (5 imágenes de un tipo de dispositivo); aun así se han logrado resultados de agrupamiento satisfactorios.

Tabla 12.8: TPR para agrupamiento asimétrico de 5 dispositivos

Grupo	**Apple Iphone 5**	**Huawei U8815**	**LG E400**	**Nokia 800 Lumia**	**Samsung GT 5830m**	**TPR**
A	50	50	50	50	50	**99,20 %**
B	100	100	100	100	100	**100 %**
C	100	95	90	85	80	**99,78 %**
D	50	45	40	35	30	**99,1 %**
E	100	75	50	25	10	**99,6 %**
F	100	30	20	10	5	**99 %**

Tabla 12.9: TPR para agrupamiento asimétrico de 7 dispositivos

Grupo	**Apple Iphone 5**	**Huawei U8815**	**LG E400**	**Nokia 800 Lumia**	**Samsung GT 5830m**	**Sony ST25a**	**Zopo Zp980**	**TPR**
A	50	50	50	50	50	50	50	**99,71 %**
B	100	100	100	100	100	100	100	**87,13 %**
C	100	95	90	85	80	75	70	**99,84 %**
D	50	45	40	35	30	25	20	**99,36 %**
E	100	75	50	25	10	5	1	**85,43 %**
F	100	50	40	30	20	10	5	**99,21 %**

Asimismo, se llevaron a cabo experimentos para probar la propuesta en un escenario donde todos los dispositivos son del mismo fabricante pero diferente modelo. Las cámaras de los móviles de un mismo fabricante deberían ser muy similares para gran parte de sus productos y, por tanto, el ruido del sensor extraído entre modelos distintos debiera asemejarse. Sin embargo, en la Tabla 12.10 podemos observar la TPR de clasificación, concluyendo que el valor de correlación entre SPNs varía lo suficiente entre los distintos modelos para poder identificar a cada uno por separado, incluso cuando los modelos son sumamente parecidos como es el caso de los modelos ST25a y ST25i.

Tabla 12.10: TPR para agrupamiento simétrico de 3 dispositivos del mismo fabricante

Modelos de Sony Ericsson	TPR para 50	TPR para 100
C2105	99,33 %	74,50 %
ST25a		
ST25i		

A pesar de que en los métodos de agrupamiento no se posee información de los conjuntos de datos a agrupar, se realizaron algunos experimentos de clasificación sobre un conjunto de 5 dispositivos diferentes con distintos conjuntos de entrenamiento tanto con distribuciones simétricas como asimétricas. En lo que se refiere estrictamente a la fase de entrenamiento se usa la técnica propuesta en [CAPI10]. De esta forma se reduce ampliamente la complejidad computacional en el momento de calcular la matriz de similitud, siendo ésta la que demora mayor tiempo en la ejecución del algoritmo.

En la Tabla 12.11 se muestra el número de imágenes utilizadas para la fase de entrenamiento. En todos los experimentos se clasificaron las 150 imágenes restantes que posee el conjunto total de imágenes de cada dispositivo. Ninguna imagen dentro del conjunto de entrenamiento se encuentra en el conjunto de clasificación y viceversa. Se puede observar que se mantienen los buenos resultados incluso con los conjuntos de imágenes asimétricos (98,3 % de TPR para los 4 experimentos realizados).

Tabla 12.11: Agrupamiento de 5 dispositivos con fase de entrenamiento

Grupo	Apple Iphone 5	Huawei U8815	LG E400	Nokia 800 Lumia	Samsung GT 5830m	TPR
A	50	50	50	50	50	98,67 %
B	100	100	100	100	100	99,33 %
C	100	75	50	25	10	96,67 %
D	50	45	40	35	30	98,53 %

En la Tabla 12.12 pueden verse los tiempos de ejecución del algoritmo de agrupamiento, sin tener en cuenta la extracción de las características, de algunos de los experimentos concretos realizados. Sólo en los casos en los que se ha realizado la fase de entrenamiento se indica el número total de imágenes utilizadas por cada dispositivo detallando cuántas son utilizadas para la fase entrenamiento y cuántas para la fase de clasificación. Con los tiempos de ejemplo de estos experimentos puede hacerse una idea global del orden en tiempo de ejecución del algoritmo. Todas las imágenes fueron recortadas a 1024×1024. El equipo donde se ejecutaron los experimentos es un Intel Core i7-2670QM 2,2 GHz con 6Gb RAM y sistema operativo Linux.

Tabla 12.12: Tiempos de ejecución

Agrupamiento	Número de Dispositivos	Número de Fotos (entrenamiento-clasificación)	Fase de Entrenamiento	Tiempo (seg)
Simétrico	3	50	No	445
		100	No	1698
	5	50	No	1207
		150 (100-50)	Sí	516
		100	No	4789
		150 (50-100)	Sí	1222
Asimétrico	5	50, 45, 40, 35, 30	No	742
		100, 95, 90, 85, 80	No	4299
	7	50, 45, 40, 35, 30, 25, 20	No	1484
		100, 95, 90, 85, 80, 75, 70	No	8992

12.4 Síntesis del Capítulo

En este capítulo se ha realizado un análisis de las principales técnicas de agrupamiento de imágenes sin supervisión en el análisis forense de imágenes digitales, realizando además una propuesta. El algoritmo de la misma se basa en la combinación de un agrupamiento jerárquico y un agrupamiento plano para la separación entre grupos. El uso del coeficiente de silueta para la validación de los grupos ha demostrado alcanzar buenos resultados al obtener altos TPRs. Además, el número de grupos ha correspondido al número de dispositivos reales en la mayoría de los casos. El porcentaje de aciertos correctos al utilizar el recorte de imagen desde la esquina izquierda obtiene mejores resultados que cuando el recorte se centra en la imagen, a pesar de encontrar diferentes referencias en la literatura argumentando la saturación y la falta de iluminación encontradas en esas regiones. Se ha experimentado con diferentes modelos de dispositivos del mismo fabricante, ya que junto con la alta tasa de aciertos correctos, mediante el uso de distribuciones simétricas y asimétricas de imágenes por dispositivo, comprobándose la adaptabilidad del algoritmo para ser aplicado en casos reales. Los experimentos realizados en este capítulo han revelado una gran diversidad de situaciones con respecto a la simetría o no de los conjuntos de imágenes, su tamaño, el número de dispositivos utilizados y el uso de dispositivos de la misma marca. Después de todos los experimentos se concluye que los resultados de la aplicación de la técnica son buenos (92,98 % de TPR en promedio para todos los experimentos).

Capítulo 13

Método Anti-Forense para Manipular la Identificación de la Fuente de Adquisición

El análisis forense de imágenes digitales surge con la idea de restaurar la fiabilidad de las imágenes digitales, que de otro modo podrían considerarse modificables muy fácilmente. Al igual que la mayoría de los campos de estudio tienen una contracorriente, y así criminales o estafadores hacen esfuerzos para manipular las imágenes en su propio beneficio. Ellos usan el conocimiento del análisis forense con el objetivo de eliminar o incluso suplantar las huellas dactilares o huellas que se utilizan para determinar la fuente de la imagen. Muchos de los algoritmos forenses en la literatura no fueron diseñados para ser robustos contra tal comportamiento y, como resultado, son fáciles de engañar. De la misma manera, los métodos de análisis forense de imágenes pueden beneficiarse de estudios sobre técnicas de ataque, con el propósito de fortalecer los algoritmos de las próximas generaciones.

Este capítulo presenta dos algoritmos para la destrucción y falsificación de la identidad de una imagen digital. Se comienza con una breve presentación de varios ataques que pueden ocurrir en el análisis forense. A continuación, se detalla el algoritmo para la destrucción de la identidad de la imagen. Posteriormente, se presenta un algoritmo para la falsificación de la identidad de una imagen. Después, para evaluar la validez de los algoritmos presentados, se exponen un conjunto de experimentos y los resultados obtenidos. El capítulo finaliza con una breve síntesis de lo expuesto en el mismo.

13.1 Generalidades

Aunque las imágenes pueden ser consideradas parte de la verdad, ya que son hechos reales captados por dispositivos electrónicos (cámaras digitales), nunca ha sido tan fácil modificar las imágenes como lo es hoy en día, dada la existencia de potentes y sofisticados programas software. Esta facilidad de manipulación plantea interrogantes sobre la integridad y veracidad de las imágenes. Las técnicas de análisis forense deben permitir detectar este tipo modificaciones malintencionadas o ser robustas ante las mismas.

En este trabajo se proponen algoritmos basados en el ruido del sensor y en la transformada wavelet que permiten eliminar la identidad de una imagen y falsificar la misma en una imagen dada. En el estado del arte existen algoritmos que realizan estas funciones con mayor cantidad y complejidad de los datos de entrada que los propuestos en este capítulo. Asimismo, necesitan tener acceso a la cámara digital a la que pertenece la imagen a la que se quiere aplicar el algoritmo. En la propuesta del algoritmo de destrucción de la identidad sólo se necesita la fotografía a anonimizar y en la propuesta de falsificación de la identidad se necesita un conjunto de fotografías de la cámara atacante y la fotografía cuya identidad se quiere falsificar, es decir, no se necesita la cámara de cuya fotografía va a ser falsificada su identidad. Estos escenarios son los más comunes y realistas, ya que en muchos casos el atacante no tiene acceso a cámara que generó la imagen a atacar. Asimismo, los algoritmos propuestos serán de ayuda al fortalecimiento de nuevas técnicas de análisis forense de imágenes de dispositivos móviles frente a posibles ataques.

En [RCAGSO+13] se concluye que el uso del ruido del sensor PRNU junto con la transformada wavelet es un método eficaz para la identificación de la fuente, alcanzando una tasa de éxito promedio del 87,21 %. Se estima que una técnica contra forense a este tipo de identificación puede estar basada en estos elementos. Por tanto, los algoritmos presentados basan su funcionamiento en el tratamiento del ruido PRNU y la transformada wavelet.

13.1.1 Algoritmo de Destrucción de la Identidad de una Imagen

En este trabajo se propone un algoritmo basado en [GFH06] que permite extraer y eliminar la huella del sensor de una imagen P_1. El algoritmo propuesto en [GFH06] obtiene un vector de características con fines de clasificación. El algoritmo propuesto en cambio tiene como fin la obtención de una imagen con su identidad destruida eliminando el ruido PRNU.

Entre los diferentes filtros que existen para la eliminación del ruido de las imágenes los que usan la transformada wavelet dan mejor resultado debido a que el ruido residual que se obtiene con este filtro contiene la menor cantidad de rasgos de la escena. Generalmente, las áreas alrededor de los bordes son mal interpretadas cuando se utilizan únicamente filtros de eliminación de ruido menos robustos, tales como el filtro de Wiener o el filtro de mediana. Por este motivo se seleccionó el filtro de eliminación de ruido basado en la transformada wavelet. Para cada nivel de descomposición wavelet se obtienen los componentes de alta frecuencia H (horizontal), V (vertical) y D (diagonal). El Algoritmo 13 muestra los pasos a seguir para eliminar la huella del sensor.

Algoritmo 13: Eliminación del PRNU

I es la imagen víctima

```
1  procedure EliminarPRNU(I)
2      Realizar una descomposición wavelet de 4 niveles de I_n;
3      foreach nivel de la descomposición wavelet do
4          foreach c ∈{H,V,D} do
5              Calcular la varianza local;
6              if varianza adaptativa then
7                  Calcular 4 varianzas con ventanas de
                    tamaños 3, 5, 7 y 9, respectivamente;
8                  Seleccionar la varianza mínima;
               else
9                  Calcular la varianza con una ventana
                    de tamaño 3;
10             Calcular los componentes wavelet sin ruido
                aplicando el filtro de Wiener a la varianza;
11     Obtener I_limpia aplicando la transformada wavelet inversa de los componentes
       limpios calculados;
12 end procedure
```

donde $I_{sinruido}$ es obtenida aplicando el Algoritmo 13 de eliminación del ruido. Cabe destacar que la $I_{sinruido}$ obtenida no es exactamente una imagen sin ningún tipo de ruido, ya que el patrón del ruido del sensor está formado por el ruido PRNU y el PNU. El Algoritmo 13 sólo elimina de la $I_{sinruido}$ el PRNU y no el PNU. Sin embargo, por facilidad en el nombrado y nomenclatura se utiliza $I_{sinruido}$ para nombrar a la imagen sin el ruido PRNU.

13.1.2 Algoritmo de Falsificación de la Identidad de la Imagen

En esta sección se propone un algoritmo para inyectar el patrón del sensor de una cámara C_1 a una imagen P_2 generada con una cámara C_2 sin requerir acceso a la cámara C_2. Para ello se utilizará el Algoritmo 13 presentado previamente en la Sección 13.1.1.

Las técnicas de identificación de la fuente basadas en PRNU calculan la huella del sensor de la imagen con la Ecuación 13.1:

$$I_{ruido} = I - I_{sinruido} \tag{13.1}$$

El patrón del ruido PNU se calcula mediante el promedio del ruido residual de varias imágenes con la Ecuación 13.2:

$$P_{ruido} = \frac{1}{N} \sum_{i=1}^{N} I_{ruido} \tag{13.2}$$

Una vez que se tiene la posibilidad de eliminar el ruido del sensor y de extraer el patrón del ruido del sensor se puede plantear la falsificación de la identidad de una imagen. El Algoritmo 14 muestra los pasos a seguir para falsificar la identidad de una imagen.

Algoritmo 14: Falsificación del PRNU

Input: I es la imagen víctima
N es el número de imágenes de superficies uniformemente iluminadas de la cámara suplantadora

① **procedure** ForgeImg(I, N)
② $I_{sinruido} \leftarrow$ RemovePRNU(I);
③ $P_{ruido} \leftarrow$ ExtractPRNU(N);
④ Realizar una descomposición wavelet de un nivel de $I_{sinruido}$ obteniendo los componentes L_I, H_I, V_I y D_I;
⑤ Realizar una descomposición wavelet de un nivel de P_{ruido} obteniendo los componentes H_P, V_P y D_P;
⑥ Calcular los componentes wavelet falsificados mediante $c_F = c_I + c_P$ donde $c \in \{H,V,D\}$;
⑦ Obtener I_{falsa} aplicando la transformada wavelet inversa con L_I, H_F, V_F y D_F;
⑧ **end procedure**

Para tener una huella de mejor calidad y obtener mejores resultados en la falsificación se recomienda que el número N de imágenes sea superior a 50 y que las imágenes se hayan adquirido de superficies planas sin textura iluminadas uniformemente. Como superficies planas se pueden considerar fotografías de cielo despejado o de un papel blanco.

13.2 Experimentos

En esta sección se describen los experimentos realizados con los algoritmos de eliminación de la huella del sensor (Algoritmo 13) y de falsificación de la identidad de una imagen (Algoritmo 14).

En estos experimentos se realizó la eliminación y la falsificación de las huellas con las implementaciones propuestas y con la herramienta "PRNU Decompare" [Net13b], la cual utiliza la técnica de corrección de sensibilidad descrita en la Sección 10.3.2.1 y permite la eliminación y la suplantación del patrón del sensor [Net13b] para llevar a cabo la eliminación y la suplantación de la huella. Esta herramienta requiere como entrada una fotografía de un marco oscuro y un número N de imágenes de superficies planas iluminadas uniformemente (se recomienda un mínimo de 30 imágenes). Los resultados obtenidos se compararon haciendo uso de la herramienta "NFI PRNU Compare" [Net13a], la cual permite comparar imágenes y patrones del ruido del sensor de varias imágenes. "NFI PRNU Compare" usa como medida para la comparación la correlación, la cual es utilizada en otros muchos trabajos ([LFG06], [GFC11], [CESR12] y [UH12], entre otros), para la comparación de imágenes y patrones de ruido. Cabe destacar que en la propuesta no se necesita un conjunto de imágenes de la cámara de cuya fotografía se quiere destruir su identidad, sólo se necesita la propia imagen. Tampoco se necesita ningún conjunto de fotografías ni tener acceso a la cámara víctima en el caso de la falsificación de la identidad, sólo un conjunto de fotografías de la cámara atacante. Es decir, el algoritmo propuesto necesita menos imágenes de entrada que "PRNU Decompare" para realizar la misma

función.

13.2.1 Destrucción de la Identidad de la Imagen

Para este experimento se utilizaron las fotografías de 6 cámaras digitales de dispositivos móviles: LG E510f, LG 400, Nokia 800 Lumia, Sony ST25i (Xperia U), Apple iPhone5 y Samsung GTI-9000. De cada uno de los dispositivos se obtuvieron 50 fotografías de imágenes planas uniformemente iluminadas y una fotografía totalmente oscura cubriendo la lente de la cámara. Una fotografía de cada cámara seleccionada al azar de la base de datos de fotografías se utiliza para la destrucción de la identidad de la imagen. Todas las fotografías fueron recortadas a un tamaño de 1024×1024.

Inicialmente, se genera el primer grupo de imágenes sin huella, haciendo uso del Algoritmo 13. Cabe remarcar que para realizar esta eliminación no se requiere de fotografías adicionales a la fotografía de la que se pretende eliminar el ruido del sensor. A continuación, se genera el segundo grupo de imágenes sin huella con la herramienta "PRNU Decompare", dando como entrada a este programa las 50 imágenes planas y la imagen del marco oscuro. Por lo tanto, se tienen dos imágenes sin la huella del sensor de cada cámara, una generada con el Algoritmo 13 y otra con "PRNU Decompare".

Para evaluar la efectividad del algoritmo de eliminación de la huella se comparan los dos grupos de imágenes con la herramienta "NFI PRNU Compare". Con 40 imágenes elegidas de forma aleatoria de las 50 de cada cámara, "NFI PRNU Compare" obtiene el patrón del ruido de cada cámara, según se indica en las recomendaciones de la documentación de la herramienta.

En la Tabla 13.1 se muestran los resultados de comparar cada patrón del ruido del sensor generado por "NFI PRNU Compare" de cada una de las cámaras con las imágenes sin ruido generadas por las dos herramientas y con la fotografía original. La herramienta "NFI PRNU Compare" permite hacer comparaciones midiendo qué tanto se parece un patrón a otro (las filas que están más cerca del patrón con el que se compara son las que más se le asemejan). La Tabla 13.1 muestra los coeficientes de correlación para cada canal de color. Una alta correlación indica un alto grado de similitud lineal entre los dos patrones.

En todos los experimentos las imágenes sin ruidos generadas con "PRNU Decompare", resultaron ser las menos parecidas al patrón. Esto era de esperar ya que consideran mayor información para eliminar la huella.

En todos los casos los resultados de las comparaciones de las imágenes sin ruido tuvieron resultados muy similares, lo que indica que en este caso el algoritmo propuesto obtuvo buenos resultados acercándose al resultado del programa "PRNU Decompare", pero sin la necesidad de usar la fotografía del marco oscuro, ni las 50 imágenes planas.

Tabla 13.1: Comparativa entre patrones e imágenes sin ruido

Patrón	Imagen	Rojo	Verde	Azul	Suma
LG E510f	Original	-0,014645672	-0,0017777978	-0,007864626	-0,024288096
	Propuesta	-0,015506644	-0,003044259	-0,008411303	-0,026962206
	Decompare	-0,018929206	-0,0023383496	-0,012027217	-0,033294775
LG E400	Original	0,011481647	0,010190065	0,01825918	0,039930895
	Propuesta	0,010315191	0,008225861	0,017940063	0,036481116
	Decompare	0,010638827	0,009045472	0,016430777	0,036115076
Nokia 800 Lumia	Original	0,011352311	0,011754888	0,019119238	0,042226437
	Propuesta	0,009912337	0,009113852	0,016991276	0,036017465
	Decompare	0,010812875	0,009995133	0,014978331	0,035786339
Apple iPhone 5	Original	-0,015712192	-0,002311284	-0,007307031	-0,025330507
	Propuesta	-0,016984729	-0,003462913	-0,009599754	-0,030047396
	Decompare	-0,019395503	-0,003140395	-0,009915681	-0,032451579
Sony ST25i	Original	-0,012013772	-0,002127295	-0,006817536	-0,020958603
	Propuesta	-0,014839721	-0,003142763	-0,009114359	-0,027096843
	Decompare	-0,016545112	-0,002611324	-0,011200112	-0,030356548
Samsung GTI-9000	Original	0,017310016	0,010754888	0,016119238	0,044184142
	Propuesta	0,014015311	0,007826601	0,014491394	0,036333306
	Decompare	0,014979864	0,007992510	0,012984029	0,035956403

Un tema importante a tratar es el de si la imagen cuya identidad se ha destruido reduce su calidad o de si existen efectos visibles al ojo humano sobre la escena de la imagen. Para ello en la Figura 13.1 se muestran dos ejemplos de imágenes de los dispositivos Nokia 800 Lumia y Sony ST25i (Xperia U) y sus respectivas imágenes con su identidad destruida. En la Figura 13.1 no se aprecian cambios en la escena de las imágenes cuya identidad se ha destruido.

Aún así, se ha decidido utilizar unas métricas IQM para evaluar de un modo objetivo la pérdida de calidad de las imágenes cuya identidad se ha destruido con respecto a las originales. Para ello se ha decidido utilizar las métricas de Minkowsky. Concretamente se utilizarán las métricas de Minkowsky para $\gamma = 1$ las cuales se corresponde con el MAE y $\gamma = 2$ con el MSE. En ambos casos, altos valores de MAE o MSE se corresponden con imágenes de baja calidad. Estas métricas son aplicadas para cada una de las bandas RGB, por lo tanto se obtendrán tres métricas para el MAE y otras tres para el MSE. A continuación, se muestran los valores de las métricas de calidad para cada una de las bandas de las 4 imágenes de la Figura 13.1. Como puede observarse en la Tabla 13.2, la destrucción de la identidad de la imagen no hacer variar prácticamente nada los índices de calidad de la imagen presentados.

(a) Nokia 800 Lumia: Imagen original

(b) Nokia 800 Lumia: Imagen con identidad destruida

(c) Sony ST25i: Imagen original

(d) Sony ST25i: Imagen con identidad destruida

Figura 13.1: Ejemplo de imágenes anonimizadas

Tabla 13.2: MAE y MSE de las imágenes y sus respectivas imágenes con identidad destruida

Imagen	MAE			MSE		
	Rojo	Verde	Azul	Rojo	Verde	Azul
Nokia 800 Lumia	0,8410	0,8183	0,8428	2,6498	2,1777	2,3869
Nokia 800 Lumia identidad destruida	0,8368	0,8135	0,8388	2,6245	2,1531	2,3617
Sony ST25i	0,8976	0,8705	0,8916	4,0664	3,3754	3,7108
Sony ST25i identidad destruida	0,8924	0,8656	0,8869	4,0324	3,3465	3,6783

13.2.2 Falsificación de la Identidad de la Imagen

Para este experimento se utilizó también el conjunto de fotografías de la Sección 13.2.1. De forma similar al experimento anterior se extrajo la huella de una de las cámaras y se inyectó en las otras dos haciendo uso del algoritmo propuesto y "PRNU Decompare". Después se compararon los resultados con la herramienta "NFI PRNU Compare". Los roles que jugaron cada una de las cámaras se muestran en la Tabla 13.3.

Tabla 13.3: Dispositivos usados para la falsificación de la identidad

Cámara Suplantadora	**Víctima 1**	**Víctima 2**
LG E510f	LG-400	Samsung GT-I8160P
Cámara Suplantadora	**Víctima 3**	**Víctima 4**
Nokia 800 Lumia	Sony ST25i	Samsung GTI-9000

Para realizar la falsificación del patrón del ruido del sensor con el algoritmo propuesto en este trabajo únicamente se requirieron las 50 imágenes planas uniformemente iluminadas pertenecientes a la cámara suplantadora. En el caso de la herramienta "PRNU Decompare" para realizar la falsificación el programa requiere como entrada las 50 fotografías planas y la fotografía totalmente oscura tanto de la cámara suplantadora como de la cámara víctima.

Después de realizar la falsificación en las dos cámaras víctimas se compararon los resultados que están resumidos en la Tabla 13.4, mostrando los coeficientes de correlación para cada banda de color.

En el caso de la víctima 1 se puede observar que las dos suplantaciones resultaron tener mayor similitud con el patrón de la cámara suplantadora y el resultado de "PRNU Decompare" se acerca más, aunque la diferencia no es muy significativa considerando que ellos utilizan un número mucho mayor de imágenes como fuente de información.

En el caso de la víctima 2 el resultado del algoritmo propuesto fue el que menos similitud tuvo con el patrón de la cámara suplantadora. Los resultados obtenidos hasta el momento eran los esperados, debido a que el algoritmo propuesto en este trabajo no asume que se tiene acceso a la cámara fuente y en el trabajo del "PRNU Decompare" sí. Aún así, en ambos casos los resultados de la propuesta son muy cercanos a los obtenidos con "PRNU Decompare".

Es importante recalcar que en escenarios reales normalmente no se tiene acceso a la cámara víctima. Esta propuesta sólo necesita un conjunto de imágenes de la cámara atacante, al igual que "PRNU Decompare". Sin embargo, esta propuesta no necesita ningún tipo de contenido especial de la escena en el conjunto de imágenes y "PRNU Decompare" sí, tal y como se ha indicado anteriormente.

Tabla 13.4: Comparativa entre patrones, imágenes originales y víctimas

Imagen		Rojo	Verde	Azul	Suma
Víctima 1	Decompare	0,009219962	0,0054620425	0,009098741	0,023780745
	Propuesta	0,007900867	0,0046529872	0,0083521	0,020905953
	Original	0,0073570074	0,004183661	0,0075896666	0,019130334
Víctima 2	Decompare	0,01418404	0,013986045	0,013574668	0,041744754
	Original	0,011300047	0,013949845	0,0125216115	0,0377715
	Propuesta	0,008964902	0,0066337977	0,004440412	0,020039111
Víctima 3	Decompare	0,00811001	0,004442147	0,009333811	0,021885968
	Propuesta	0,007201833	0,00419231	0,009112659	0,020506802
	Original	0,006232409	0,003880702	0,007871138	0,017984249
Víctima 4	Decompare	0,009913582	0,007213382	0,008788015	0,025914979
	Propuesta	0,008271773	0,007621801	0,008141919	0,024035493
	Original	0,008137003	0,005338104	0,00790911	0,021384217

Análogamente al caso de destrucción de la identidad de la imagen, se realiza el mismo estudio de pérdida de calidad de la imagen para el ejemplo de falsificación de las víctimas 3 y 4. En la Tabla 13.5 se muestran los índices de calidad de la imagen MAE y MSE para las imágenes originales y sus respectivas imágenes con la identidad falsificada.

Tabla 13.5: MAE y MSE de las imágenes y sus respectivas imágenes con identidad falsificada

Imagen	MAE			MSE		
	Rojo	Verde	Azul	Rojo	Verde	Azul
Víctima 3 original	0,8976	0,8705	0,8916	4,0664	3,3754	3,7108
Víctima 3 falsificada	0,8926	0,8656	0,8872	4,0323	3,3445	3,6787
Víctima 4 original	0,7836	0,7800	0,7820	2,3412	2,2525	2,2724
Víctima 4 falsificada	0,7685	0,7639	0,7661	2,2940	2,2062	2,2207

Como puede observarse en la Tabla 13.5, la falsificación de la identidad de la imagen no hace variar prácticamente nada los índices de calidad de la imagen presentados. Los resultados pueden verse en la Figura 13.2, donde puede comprobarse que los cambios son inperceptibles para el ojo humano.

(a) Víctima 3 original

(b) Víctima 3 con identidad falsificada

(c) Víctima 4 original

(d) Víctima 4 con identidad falsificada

Figura 13.2: Ejemplo de imágenes falsificadas

13.3 Síntesis del Capítulo

En este capítulo se han propuesto dos nuevos algoritmos, uno capaz de eliminar todos los datos que permiten encontrar la identidad de una imagen y otro que permite su falsificación. Los dos algoritmos se basan en el uso de diferentes tipos de ruido del sensor y en la transformada wavelet. Ambos algoritmos tienen una gran ventaja sobre otros existentes ya que necesitan una cantidad de datos de entrada mucho menor y además éstos son más fáciles de obtener, lo que los hace más aplicables y más cercanos a los escenarios de la vida real.

Ambos algoritmos pueden ser vistos como contribuciones de investigación al futuro fortalecimiento de las técnicas forenses para detectar manipulaciones malintencionadas. Por ejemplo, el primer algoritmo puede ser muy útil en aplicaciones web que cargan y

muestran imágenes en Internet (redes sociales, directorios, etc.), ya que permiten la carga de imágenes totalmente anónimas.

La eficacia de los algoritmos propuestos es buena, aunque en algunos casos no se obtienen resultados comparables con los de otros algoritmos, logrando generalmente resultados cercanos y bastante aceptables, con el beneficio de reducir drásticamente los datos de entrada requeridos y ser más prácticos y realistas. Como se ha comentado anteriormente, estos algoritmos pueden ser útiles como punto de partida para futuras mejoras que permitan obtener resultados similares mediante otros algoritmos o herramientas, enfatizando y teniendo en cuenta los limitados datos de entrada necesarios y que puedan funcionar incluso cuando no tengan acceso a la cámara de la víctima en el caso de falsificación de la identidad de una imagen. Además, la aplicación de ambos algoritmos no provoca cambios visibles en la imagen y no reduce significativamente la calidad de la misma.

Capítulo 14

Conclusiones y Trabajo Futuro

Numerosas imágenes digitales circulan diariamente en Internet o se utilizan como prueba en los procedimientos judiciales. Como consecuencia, el análisis forense de las imágenes digitales generadas por dispositivos como cámaras digitales, dispositivos móviles, escáneres o computadores se vuelve importante en diversas situaciones de la vida real. Cabe destacar que las técnicas de análisis forense de imágenes generadas por dispositivos móviles son, por sus características intrínsecas, específicas para éstos, no siendo válidas en la mayoría de los casos las técnicas clásicas de análisis forense de imágenes. Un ejemplo de esto se presenta cuando un analista forense necesita identificar el tipo de dispositivo que generó una imagen (cámara, escáner, computador) o la clase (marca y modelo) de la fuente de adquisición de la imagen.

En este trabajo se ha presentado una técnica para identificar imágenes de dispositivos móviles con respecto a imágenes escaneadas y generadas por computador. Además, se ha introducido otra técnica que permite distinguir la fuente de adquisición de imágenes de dispositivos móviles. Las técnicas se basan en el uso de cuatro conjuntos de características (Ruido, Color, IQM y Wavelets), con los que se han realizado diferentes ajustes con el fin de mejorar los resultados para cada tipo de dispositivo específico. Se ha experimentado con la combinación de las distintas características, tamaño, posición de recorte y funciones wavelets. La clasificación se ha realizado con SVMs.

Con respecto a la identificación del tipo de fuente, la primera conclusión general es que las características basadas en el ruido se descartan como inválidas cuando el número de tipos de dispositivos es mayor que 2. Esto se debe a que en los experimentos se obtuvieron resultados inaceptables en la identificación con tres tipos de dispositivos (escáner, dispositivo móvil y computador). Los experimentos en los que se utilizaron imágenes enteras y diferentes tamaños de recorte y posiciones arrojaron también resultados inaceptables al identificar tres tipos de dispositivos (escáner, dispositivo móvil y computador). Como se discute en los experimentos, para estos tres tipos de dispositivos hay docenas de fabricantes y modelos distintos que dificultan la clasificación.

Como contrapartida, los analistas forenses sí pueden tener en cuenta la aplicación de la técnica con características de ruido para la identificación del tipo de fuente de imágenes de dispositivos móviles con respecto a imágenes de escáneres y computadores. Los resultados son mucho mejores al discernir el tipo de fuente entre escáneres y dispositivos móviles.

El uso de todas las características mejora significativamente los resultados, pero como conclusión general no son lo suficientemente buenos para ser utilizados en una situación comprometida.

Los resultados son mucho más alentadores al identificar la fuente de adquisición de imágenes de dispositivos móviles. En todos los conjuntos de experimentos realizados hay por lo menos una configuración que produce buenos resultados, poniéndolos siempre en el contexto del nivel de demanda de esta técnica (un gran número de dispositivos o muchos dispositivos del mismo fabricante).

El uso de todas las características o las características wavelets de la familia Daubechies 8-tap son las que ofrecen mejores resultados. Con respecto al tamaño de recorte, para la obtención de resultados óptimos existe un tamaño de recorte óptimo (1024×768), que no necesariamente es el mayor o la imagen entera, ya que ésta última genera peores resultados que cuando se utiliza un recorte. Cuando se considera un recorte de tamaño suficientemente grande, por ejemplo 1024×768, la reducción del recorte disminuye la tasa de éxito. Con respecto al número de dispositivos utilizados, como se esperaba, cuanto mayor es el número de dispositivos, menor es la tasa de éxito. Lo mismo ocurre cuando se utilizan dispositivos del mismo fabricante, cuyas cámaras son similares o idénticas en algunos casos.

Por lo tanto, el analista forense, conociendo a priori la información en algunos casos y teniendo en cuenta estas conclusiones, debe decidir establecer los diversos parámetros de la técnica y la validez de los resultados, teniendo en cuenta los porcentajes obtenidos en los experimentos presentados en este trabajo. En otras palabras, una sola aplicación de la técnica puede dar buenos resultados en algunos casos y malos resultados en otros, dependiendo de factores tales como si se quiere decidir el tipo o marca y modelo de los dispositivos, el número de dispositivos utilizado o el número de dispositivos del mismo fabricante, entre otras cosas.

Los técnicas de identificación de la fuente que usan clasificación con SVMs en donde es requisito conocer a priori las clases a las que pertenecen las imágenes en algunos escenarios no son aplicables. Por ejemplo, en escenarios en los que el analista desconoce por completo el conjunto de cámaras a las que pueden pertenecer un conjunto de imágenes. Así surgen otras técnicas basadas en agrupamiento. En este tipo de clasificación no se tienen datos de las cámaras a priori y el objetivo no es identificar la marca y modelo de la cámara, sino poder agrupar distintas imágenes en grupos disjuntos en los que todas sus imágenes pertenecen al mismo dispositivo.

En este trabajo también se ha realizado un análisis de las principales técnicas de agrupamiento de imágenes sin supervisión, siendo éstas de suma importancia en el análisis forense de imágenes digitales. A pesar del auge que han tenido las cámaras de dispositivos móviles en estos tiempos, aún no existen en el estado del arte muchas referencias para el agrupamiento no supervisado de imágenes de dispositivos móviles. La mayor parte de los trabajos se refieren a la clasificación supervisada y en muchos casos no se centran en imágenes de dispositivos móviles, las cuales tienen características peculiares.

La comparación de los resultados de este trabajo con los de otros trabajos del estado del arte no puede realizarse de forma precisa, ya que en los mismos no se hace referencia al número final de grupos o grupos generados, lo cual es fundamental. Además, en estos

trabajos no se detalla cómo se han calculado las tasas de acierto, ni se hace referencia a las mismas cuando los grupos generados por la clasificación son diferentes en número a la cantidad de dispositivos utilizados, haciendo esto que la comparativa de sus tasas con respecto a la interpretación del TPR realizada carezca de sentido. El ruido agregado en cada imagen por el sensor de la cámara, debido a los fallos en el proceso de fabricación de éste o defectos por el uso diario, ha demostrado ser una fuente fiable de identificación de un dispositivo. Asimismo, el cálculo de correlación normalizada entre ruidos de sensor extraídos de dos o más imágenes es una medida de similitud bastante utilizada en las técnicas de aprendizaje sin supervisión de imágenes, siendo las técnicas de agrupamiento aquellas que tienen mejores resultados.

El algoritmo propuesto para el agrupamiento está basado en la combinación de un agrupamiento jerárquico y un agrupamiento plano para la separación entre grupos. El uso del coeficiente silueta para la validación de los grupos demostró dar buenos resultados al obtener elevados TPRs. También el número de grupos correspondió al número de dispositivos reales en la mayoría de los casos.

El porcentaje de aciertos al utilizar el recorte de la esquina superior izquierda de la imagen fue mayor que al recortar la imagen por el centro, pese a encontrar diferentes referencias en la literatura argumentando la saturación y ausencia de iluminación encontrada en esas regiones.

También fue importante haber experimentado con diferentes modelos de dispositivos del mismo fabricante, ya que, junto con la alta tasa de aciertos correctos mediante el uso de distribuciones simétricas y asimétricas de imágenes (mismo o diferente número de imágenes) por dispositivo, se comprueba la adaptabilidad del algoritmo a su aplicación en casos reales.

Los experimentos realizados en este trabajo han permitido comprobar gran diversidad de situaciones con respecto a la simetría o no de los conjuntos de imágenes, su tamaño, el número de dispositivos utilizados y el uso de dispositivos de la misma marca. Después de todos los experimentos realizados, se concluye que los resultados de la aplicación de la técnica son buenos (92,98 % de TPR promedio de todos los experimentos realizados).

Es importante tener en cuenta los distintos ataques que pueden sufrir las técnicas propuestas con el fin de mejorarlas. Así, se han presentado dos algoritmos, uno que permite destruir la identidad de una imagen y otro que posibilita falsificar la misma. Los dos algoritmos tienen como base la utilización del ruido del sensor y la transformada wavelet.

Ambos algoritmos tienen como gran ventaja con respecto a otros con los mismos fines que necesitan una menor variedad y cantidad de datos de entrada, ajustándose en mayor medida a escenarios reales. Concretamente, para el algoritmo de eliminación de la huella de una imagen se necesita únicamente la propia imagen y no un conjunto de imágenes planas y una imagen del marco oscuro de la cámara víctima como ocurre en el caso de otros enfoques. Los algoritmos que requieren numerosas imágenes digitales con características especiales para su funcionamiento no son realistas, ya que difícilmente se puede tener acceso a la cámara víctima. Para el caso del algoritmo propuesto de falsificación de la identidad de una imagen tampoco se necesita tener acceso a la cámara víctima.

Los dos algoritmos pueden considerarse útiles para futuros fortalecimientos de las

técnicas forenses de detección de manipulaciones malintencionadas. Por ejemplo, el primer algoritmo puede ser de gran utilidad en aplicaciones web que permiten subir y mostrar imágenes en Internet (redes sociales, directorios de imágenes, etc.), ya que permite que la imagen subida sea anónima desde el punto de vista de la identificación de la fuente de adquisición.

La eficacia de estos algoritmos es bastante buena, incluso en algunos casos en los que no se obtienen resultados comparables con otros algoritmos. En general, se logran resultados cercanos y aceptables, con el beneficio de reducir drásticamente la cantidad de datos de entrada y siendo más prácticos y realistas.

Estos algoritmos pueden ser de utilidad como punto de partida para futuras mejoras que permitan obtener resultados similares a otros algoritmos o herramientas, enfatizando la mínima cantidad de datos de entrada necesarios para su funcionamiento y el hecho de no requerir acceso a la cámara víctima. Además, la aplicación de ambos algoritmos no provoca cambios visibles o degradación en la imagen, no reduciendo significativamente la calidad de la misma.

14.1 Trabajo Futuro

La información de estas conclusiones puede ser un punto de partida para futuros trabajos, como los que se señalan a continuación:

- **Búsqueda de nuevos conjuntos de características para la identificación de la fuente de adquisición de imágenes**: Interesa encontrar nuevas características y configuraciones que permitan mejorar por sí solas o conjuntamente con las presentadas en este trabajo las tasas de acierto o presenten nuevas alternativas para distintas aplicaciones. En especial, para los casos en los que la tasa de acierto tiene que ser muy cercana al 100 % como en aplicaciones relacionadas con casos judiciales. De la misma forma, cuando el número de dispositivos es muy grande, para así poder aplicar los algoritmos a grandes bases de datos de análisis forense.

- **Mejorar los resultados de las técnicas de agrupamiento**: Interesa optimizar el número de clases generadas por la técnica, es decir, que en todos o en la inmensa mayoría de los casos sea igual el número de dispositivos a clasificar al número de conjunto creados por el algoritmo de agrupamiento. Una vez logrado este objetivo, convendría optimizar la homogeneidad de los grupos, evitando clases generadas por la técnica que contengan objetos de distintos dispositivos.

- **Agrupamiento basado en características de la imagen**: El uso de otras características de la imagen diferentes a las del SPN para clasificar las imágenes, pero siguiendo el mismo algoritmo de agrupamiento, podría mejorar o darle nuevas aplicaciones al agrupamiento. Así, podrían analizarse las características de wavelets, color y calidad de imagen (IQM), entre otras.

- **Agrupamiento para vídeos**: Adaptación de las técnicas de agrupamiento a vídeos generados por dispositivos móviles.

- **Mejora de la robustez de las técnicas de análisis forense**: Mejorar las técnicas presentadas (y, por ende, cualquier técnica futura) para que sean más fuertes con respecto a los distintos posibles ataques. Este aspecto no ha sido tenido en cuenta en la creación de las distintas técnicas presentadas, ya que el principal objetivo ha sido obtener buenos resultados para imágenes no manipuladas.
- **Optimización en la destrucción de la identidad de imágenes**: La destrucción de la identidad de una imagen no debe ser considerada únicamente como un ataque. Imágenes en la web en muchos casos pueden requerir de esta función. Por tanto, se debe de investigar cómo poder eliminar todas o la mayor parte de cualquier tipo de traza que permita la identificación de una imagen, siempre teniendo como premisa que no se pierda calidad de la imagen.

Part III

Papers Related to This Thesis

Chapter 15

List of Papers

1. Ana Lucila Sandoval Orozco, Jocelin Rosales Corripio, David Manuel Arenas González, Luis Javier García Villalba, Julio César Hernández Castro: Techniques for Source Camera Identification. Proceedings of the 6th International Conference on Information Technology (ICIT 2013), Amman, Jordan, May 8 – 10, 2013.

2. Jocelin Rosales Corripio, David Manuel Arenas González, Ana Lucila Sandoval Orozco, Luis Javier García Villalba, Julio Hernandez-Castro, Stuart James Gibson: Source Smartphone Identification Using Sensor Pattern Noise and Wavelet Transform. Proceedings of the 5th International Conference on Crime Detection and Prevention (ICDP 2013), pages 1–6, London, UK, December 16 – 17, 2013.

3. Jocelin Rosales Corripio, David Manuel Arenas González, Ana Lucila Sandoval Orozco, Luis Javier García Villalba: Identificación de la Fuente de Imágenes de Dispositivos Móviles Basada en el Ruido del Sensor. Actas de la XIII Reunión Española sobre Criptología y Seguridad de la Información (RECSI 2014), pages 277–280, Alicante, Spain, September 2 – 5, 2014.

4. Luis Javier García Villalba, Ana Lucila Sandoval Orozco, Jocelin Rosales Corripio: Smartphone Image Clustering. Expert Systems with Applications, 42 (4): 1927–1940, March 2015.

5. Jocelin Rosales Corripio, Ana Lucila Sandoval Orozco, Luis Javier García Villalba: Unsupervised Classification of Mobile Device Images. Proceedings of the 7th International Conference on Information Technology (ICIT 2015), pages 96–101, Amman, Jordan, May 12 – 15, 2015.

6. Jocelin Rosales Corripio, Ana Lucila Sandoval Orozco, Luis Javier García Villalba: New Technique of Forensic Analysis for Digital Cameras in Mobile Devices. Proceedings of the 7th International Conference on Information Technology (ICIT 2015), pages 597–602, Amman, Jordan, May 12 – 15, 2015.

7. Ana Lucila Sandoval Orozco, Luis Javier García Villalba, David Manuel Arenas González, Jocelin Rosales Corripio, Julio César Hernández Castro, Stuart Gibson:

Smartphone Image Acquisition Forensics Using Sensor Fingerprint. IET Computer Vision, 9 (5): 723–731, October 2015.

8. Jocelin Rosales Corripio, Anissa El-Khattabi, Ana Lucila Sandoval Orozco, Luis Javier García Villalba: Método Anti-Forense para Manipular la Fuente de Adquisición de una Imagen de Dispositivo Móvil. Actas del VIII Congreso Iberoamericano de Seguridad Informática (CIBSI 2015), pages 176–182, Quito, Ecuador, November 10 – 12, 2015.

9. Ana Lucila Sandoval Orozco, Jocelin Rosales Corripio, Luis Javier García Villalba, Julio César Hernández Castro: Image Source Acquisition Identification of Mobile Devices Based on the Use of Features. Multimedia Tools and Applications, 75 (12): 7087–7111, June 2016.

10. Jocelin Rosales Corripio, Anissa El-Khattabi, Ana Lucila Sandoval Orozco, Luis Javier García Villalba: Theia: Una Herramienta para el Análisis Forense de Imágenes Digitales de Dispositivos Móviles. Actas de la XIII Reunión Española sobre Criptología y Seguridad de la Información (RECSI 2016), pages 226–231, Mahón, Spain, October 26 – 28, 2016.

11. Jocelin Rosales Corripio, Esteban Alejandro Armas Vega, Ana Lucila Sandoval Orozco, Luis Javier García Villalba: Uso de Características en la Identificación de la Fuente de Imágenes de Dispositivos Móviles. Actas de la XIII Reunión Española sobre Criptología y Seguridad de la Información (RECSI 2016), pages 232–237, Mahón, Spain, October 26 – 28, 2016.

12. Ana Lucila Sandoval Orozco, Jocelin Rosales Corripio, David Manuel Arenas González, Luis Javier García Villalba, Julio César Hernández Castro: Theia: A Tool for the Forensic Analysis of Mobile Devices Pictures. Computing, 98 (12): 1251–1286, December 2016.

13. Luis Javier García Villalba, Ana Lucila Sandoval Orozco, Jocelin Rosales Corripio, Julio César Hernández Castro: A PRNU-based Counter-Forensic Method to Manipulate Smartphone Image Source Identification Techniques. Future Generation Computer Systems (In Press), DOI:10.1016/j.future.2016.11.007.

TECHNIQUES FOR SOURCE CAMERA IDENTIFICATION

Ana Lucila Sandoval Orozco[1], Jocelin Rosales Corripio[1], David Manuel Arenas González[1], Luis Javier García Villalba[1] Julio César Hernández Castro[2]

[1] Group of Analysis, Security and Systems (GASS)
Department of Software Engineering and Artificial Intelligence (DISIA)
School of Computer Science, Office 431, Universidad Complutense de Madrid (UCM)
Calle Profesor José García Santesmases s/n, Ciudad Universitaria, 28040 Madrid, Spain
Email: {asandoval, darenas, javiergv}@fdi.ucm.es, jocelinr@estumail.ucm.es

[2] School of Computing, Office S129A, University of Kent
Cornwallis South Building, Canterbury CT2 7NF, UK
E-mail J.C.Hernandez-Castro@kent.ac.uk

Abstract

Digital image forensics has lately become one of the very important applications to identify the characteristics and the originality of the digital devices. This paper studies the recent developments in the field of image source identification. Proposed techniques in the literature are categorized into five primary areas based on source model identification: Metadata, Image Features, CFA and Demosaicing Artifacts, Lens Distortions and Wavelet Transforms. The main idea of the proposed approaches in each category is described in detail, and reported results are discussed to evaluate the potential of the methods.

Keywords - Image forensics, source camera identification, classification, SVM.

1 INTRODUCTION

Image source identification research investigates the design of techniques to identify the characteristics of digital data acquisition device (e.g., digital camera and cell-phone) used in the generation of an image. These techniques are expected to achieve two major outcomes. The first is the class (model) properties of the source, and the second is the individual source properties.

The success of image source identification techniques depends on the assumption that all images acquired by an image acquisition device will exhibit certain characteristics that are intrinsic to the acquisition devices because of their (proprietary) image formation pipeline and the unique hardware components they deploy, regardless of the content of the image. (It should be noted that such devices generally encode the device related information, like model, type, date and time, and compression details, in the image header, e.g., EXIF header. However, since this information can be easily modified or removed, it cannot be used for forensics purposes).

1.1 Image Formation in Digital Cameras

The design of image source identification techniques requires an understanding of the physics and operation of these devices. The general structure and sequence of stages of image formation pipeline remains similar for almost all digital cameras, although much of the details are kept as proprietary information of each manufacturer.

Consumer level digital cameras consist of a lens system, sampling filters, colour filter array, imaging sensor and a digital image processor [1]. The lens system is essentially composed of a lens and the mechanisms to control exposure, focusing, and image stabilization to collect and control the light from the scene. After the light enters the camera through the lens, it goes through a combination of filters that includes at least the infra-red and anti-aliasing filters to ensure maximum visible quality. The light is then focused onto imaging sensor, an array of rows of columns of light-sensing elements called pixels. Digital cameras deploy charge-coupled device (CCD) or complimentary metal-oxide

semiconductor (CMOS) type of imaging sensors. Each light sensing element of sensor array integrates the incident light over the whole spectrum and obtains an electric signal representation of the scenery. Since each imaging sensor element is essentially monochromatic, capturing colour images requires separate sensors for each colour component. However, due to cost considerations, in most digital cameras, only a single sensor is used along with a colour filter array (CFA). The CFA arranges pixels in a pattern so that each element has a different spectral filter. Hence, each element only senses one band of wavelength, and the raw image collected from the imaging sensor is a mosaic of different colours and varying intensity values. The CFA patterns are most generally comprised of red-green-blue (RGB) and cyan-magenta-yellow (CMY) colour components. The measured colour values are passed to a digital image processor which performs a number of operations to produce a visually pleasing image. As each sub-partition of pixels only provide information about a number of colour component values, the missing colour values for each pixel need to be obtained through a demosaicing operation. This is followed by other forms of processing like white point correction, image sharpening, aperture correction, gamma correction and compression. Although the operations and stages explained here are standard stages in a digital camera pipeline, the exact processing detail in each stage varies from one manufacturer to the other, and even in different camera models manufactured by the same company.

2 SOURCE MODEL IDENTIFICATION

The features that are used to differentiate camera-models are derived based on the differences in processing techniques and the component technologies.

The deficiency of this methodology, in general, is that many models and brands use components by a few manufacturers, and processing steps/algorithms remain the same or very similar among different models of a brand. Hence, reliable identification of a source camera-model depends on characterization of various model dependent features as explained below.

2.1 Techniques Based on Metadata

These are the simplest although they strongly depend on the data the maker inserts as metadata when the picture is taken. Furthermore, this method is the most vulnerable to malicious changes by third parties. Nevertheless, once it is proven that there is no kind of external modification, analysing the large amount of metadata can greatly help the forensic analyst.

There are a huge amount of papers referencing the different types of metadata in pictures for search and classification purposes [2, 3, 4, 5]. As stated before, these kinds of techniques, though simplest, depend on the metadata the maker may introduce. In fact, the most followed specification to identify the source of the camera, Exif [6], has two specific tags: “Make”' and “Model”, unfortunately filling data in those tags is not mandatory.

2.2 Techniques Based on Image Features

Tsai et al in [7] proposed approach methods to determine source camera or mobile phone with camera. They used a set of image features to find out about the characteristics of the camera. The features include colour features, quality Features and Image Characteristics of frequency domain. They adopt the Wavelet Transform method for calculating wavelet domain statistics and add the SVM optimal parameter setting to search step to enhance the identification rate of their previous research. The results obtained over four cameras models from two different camera brands yielded average accuracies close to 92%.

McKay et al in [8] extends *Image Source Identification* to device types such as cell phones cameras, digital cameras, scanners and computer-graphics. To achieve this, firstly they should find sources of variation among different types of devices and between different models of a device. This can be done using the dissimilarities in the image acquisition process of the imaging devices to develop two groups of features, namely colour interpolation coefficients and the noise features. They can also use these features to obtain a correct identification. Their experiments used five different models of cell phone, five models of digital cameras and four scanner models to identify the source type. The overall results were an identification accuracy of 93.75%. In their analysis of the identifying device brand/model of cell phone, obtained accuracy close to 97.7% for five models.

Jiang et al in [9] point out the fact that different patterns of sensor noise have been used for source identification successfully. However, these techniques present a particular problem since most of the time once the photos have been obtained are reprocessed, e.g., rescaling, cropping, compressing, etc. The image modifications generally destroy the fingerprint that the sensor noise could leave invalidating the sensor noise based approaches.

As a result of the previously mentioned issue, the authors propose a method that employs the marginal density DCT (Discrete Cosine Transform) coefficients in low-frequency coordinates and neighbouring joint density features on both intra-block and inter-block from the DCT domain. Additionally, they use hierarchical clustering and SVM with linear and RBF kernel to distinguish the smartphone source and processing operations applied. They experimented with different scale factors images belonging to five different smartphone models from four manufactures, obtaining a mean testing accuracy between 86.36% and 99.91%, and achieving better results while using linear kernel. Despite these satisfactory results, they could be enhanced by optimizing the kernel parameters, increasing de image data set and adopting a sophisticated feature selection algorithm.

2.3 Techniques Based on CFA and Demosaicing Artifacts

The choice of CFA and the specifics of the demosaicking algorithm are some of the most pronounced differences among different digital camera-models. In digital cameras with single imaging sensors, the use of demosaicking algorithms is crucial for correct rendering of high spatial frequency image details, and it uniquely impacts the edge and colour quality of an image. Essentially, demosaicking is a form of interpolation which in effect introduces a specific type of inter-dependency (correlations) between colour values of image pixels. The specific form of these dependencies can be extracted from the images to fingerprint different demosaicking algorithms and to determine the source camera-model of an image. Brayman et al in [1], describe their approaches to identify, detect and classify traces of demosaicking operation. They rely on two methods: The first method is based on the use of Expectation-Maximization algorithm which analyses the correlation of each pixel value to its neighbours; the second method is based on analysing inter-pixel differences. They divide their experiments into two categories. The first category of experiments was performed to assess the accuracy of camera-model identification method and the second category of experiments evaluated the improvement in the accuracy of individual camera identification method.

The accuracy in identifying the source of an image among four and five camera-models is measured as 88% and 84.8%, respectively, using images captured under automatic settings and at highest compression quality levels.

In [10], Çeliktutan et al use a set of Binary similarity measures, which are the metrics used for measuring the similarity between the bit-planes of an image. The underlying assumption is that proprietary CFA interpolation algorithm leaves correlations across adjacent bit-planes of an image that can be represented by these measures. 108 binary similarity measures are obtained for image classification purpose. The results of your experiment for a group of 9 cameras has accuracy is only 62% collecting 200 images from each one of the maximum resolution, size of 640x480 pixels, at day light and auto-focus mode.

2.4 Techniques Based on the Use of Sensor Imperfection

They can be divided into two large branches: pixel defects or sensor noise patterns.

Geradts et al [11] examine CCD pixel defects but it is not fully relevant in our case (CMOS). This technique includes point defects, hot points, dead pixel, pixel traps and cluster defects. The result noted that each one of the cameras had a different defect pattern. Nevertheless, it also noted that the number of defects in the pixels for a camera differed between pictures and varies greatly depending on the content of the image. It was also revealed that the number of defects varied at different temperatures. Finally, the study found that cameras with high-end CCD did not have this kind of problem, meaning that not all cameras suffered from this issue. It is also true that most cameras have additional mechanisms to compensate for this kind of problem.

In [12] Luka et al propose a method based on the non-uniformity of the pixels (PNU Pixel Non-Uniformity), which is a great source for the retrieval of noise patterns, which allows identifying the sensors and therefore the camera. The result for pictures with different sizes and cropped images is not satisfactory [13].

Costa et al [14] Postulate an approach for source camera attribution considering an Open Set scenario, which means that it cannot be taken for granted a full access to all possible source cameras. This proposal comprises three strands: definition of regions of interest, feature characterization, and source camera attribution. Different regions of the images can contain different information about the fingerprint of the source camera. This approach in contrast to others considers different areas of interest and not just the central region of the image. For each image, nine regions of interest (ROI) are defined as illustrated in Fig 1.

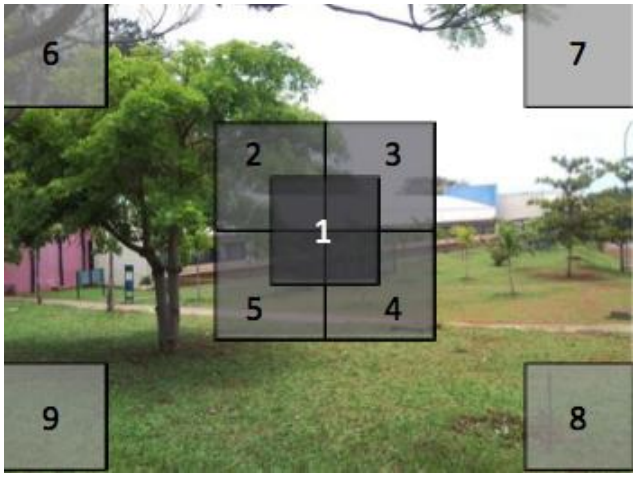


Fig. 1. Regions of interest [14]

It is assumed that these regions coincide with the principal axis of the lens and should have more scene details because amateur photographers usually focus the object of interest in the centre of the lens. Moreover regions 6 to 9 provide important information because some cameras have an effect generated by the vignetting, meaning a radial intensity downfall from the centre of the image which causes a loss of brightness or saturation at the periphery.

An important aspect to note from this kind of region characterization is that it allows comparing images with different resolutions without colour interpolation artefacts, and it is not necessary to do zero-padding, for instance, when comparing images of different sizes.

For the purpose of obtaining a feature characterization, they compute the sensor pattern noise considering the R, G, and B channels separately. In addition, they calculated the SPN for the Y channel (luminance, from YCbCr colour space) which is a combination of R, G and B channels (as a gray scale version of the image). A feature vector is formed considering the correlation between each ROI yielding in a total of 36 features to represent each image; afterwards images taken by the camera under investigation are labelled as the positive class and the remaining available cameras as the negative classes.

Finally, they came forward with a proposal to solve the source attribution problem in an open set scenario. A SVM with a RBF kernel is used to find a classifier from the training set of examples considering the positive and the available negative samples.

They take into consideration the unknown classes by moving the decision hyper plane by a value inwards to the positive class or outwards in the direction of the negative known class(es), in this way they can vary how strict they want to be in order to determine if an image belongs to a class or not. They loosely call this process as Decision Boundary Carving (DBC).

In their experiments they use a dataset with 25 digital cameras from 9 manufacturers, 150 images in JPEG format were generated for each camera with different configurations of light, zoom and flash. They achieved 94.49%, 96.77%, and 98.10% of accuracy using open sets with 2/25, 5/25, and 15/25 cameras respectively. Defining an open set with x/y as the set of y cameras where x cameras are used for training and for testing the images can belong to any of the x known cameras as well as to the other y-x unknown cameras.

2.5 Techniques Based on Wavelet Transforms

In Meng et al [15] proposes a feature-based method for source camera identification. This method employs the magnitude and phase statistics of bi-coherence along with wavelet coefficient statistics,

focusing on capturing the unique non-linear distortions on higher-order image statistics produced by different cameras and the impact of image processing operations on the wavelet domain.

First, in order to obtain the non-linear distortions characterization Bi-Coherence Features are extracted: The normalized bi-spectrum of the signal is estimated by dividing the signal into N (possibly overlapping) segments, computing the Fourier transform of each segment, and averaging the individual estimates. The mean of the magnitude and the negative phase entropy of the bi-coherence are computed as statistic features.

For reducing memory and the computational overhead implicated when calculating the total four-dimensional bi-coherence of the images, they decided to restrict their analysis to one-dimensional row, column and radial slices through the centre of the images. It is interesting to note that no rigorous constraints are placed on image sample selection since when applying bi-coherence statistics it is not necessary to extract information associated with image content (e. g., line segments).

Next, Four-scale wavelet decomposition is employed to split the frequency space into four scales and orientations. Then, four statistics (mean, variance, skewness and kurtosis) of the subband coefficients and the linear prediction errors at each orientation, scale and colour channel are computed. These statistics compose the second group of statistical feature vectors used for source camera identification. Once the bi-coherence and wavelet statistics are computed, the sequential forward featured selection (SFFS) algorithm [16] is used to reduce the correlation among features and computing load, while keeping the same classification accuracy. The SFFS method analyses all the features and builds the most significant set from them by adding and removing features until no more improvements are available.

Finally, the most representative features are classified by multi-class SVM using a C-support vector classification with non-linear RBF kernel with two tuneable parameters.

They performed experiments under the following conditions: six different model cameras from four manufactures, image of different resolutions, JPEG format, and a total of 2,100 images obtained from typical shots varying from nature scenes to close-ups of people (350 images from each camera).

As a result, they obtained a noteworthy average identification accuracy that exceeds the 97% distinguishing different models of the same brand. However, further improvements could be made by incorporating features from other techniques such as the following approach.

Wang et al [17] Describe an approach to source camera identification extracting and classifying wavelet statistic features, this method is mainly composed of three phases: Wavelet Features Extraction, Wavelet Features Selection, and Wavelet Feature Classification.

Outstanding features of wavelets domain are extracted integrating the statistical model for natural digital image from the wavelet coefficients including 216 higher-order wavelet features and 135 wavelet coefficient co-occurrence statistics. Being considered as the most significant in the identification process, features from the wavelet domain are preferred over spatial features (image color and Image Quality Metrics IQM) and Colour Filter Array (CFA). Analogously to the foregoing method, Four-scale wavelet decomposition is employed based on Separable Quadrate Mirror Filters (QMFs) to split the frequency space, the same four statistics (mean, variance, skewness and kurtosis) and the linear prediction errors are extracted.

The statistics above do not concern the texture correlation, as it has been observed that the co-occurrence features are the best among those used in the image texture feature extraction [18]. Hence, in order to take into account the texture correlation between the wavelet coefficients a co-occurrence matrix is constructed from those coefficients to form an image texture representation and distance calculation is applied in the same orientation to coefficients of co-occurrence matrix between different scales. Then statistical features (energy, entropy, contrast, homogeneity and correlation) are calculated from those distances. The wavelet feature selection and classification processes are performed in the same manner as the above method, using SFFS algorithm to select the most representative features and a multi-class SVM with the non-linear RBF kernel as a classifier.

Under the same conditions as in their prior experiments they succeeded in distinguishing different models of the same camera brand and besides, they increased their past accuracy average to a 98%. This improvement might be due to the consideration of texture features, minimizing the negative effects found in the classifier training when using multiple resolutions in images of the same model and brand. Despite of this result, improvements could still be made by evaluating the robustness of

the identification system proposed for the feature vector, and also by extending the image data set in favour of covering more brands, models, textures and contents.

Ozparlak and Avcibas in [19] Exposes a differentiating images technique using transforms from the wavelet family. They propose statistical models for ridgelet, and contourlet subbands.

1. **Ridgelet Transform**: Wavelets perform well at catching zero-dimensional or point singularities. Nevertheless, two-dimensional signals (i.e., images) normally contain one-dimensional singularities (i.e., edges and corners). In order to overcome the above mentioned drawbacks of wavelet, the system called “ridgelets” was developed. The main idea is to use Radon Transform (RAT) to map the line singularities to point singularities. Then, the mapped point singularities in the Radon domain can be effectively handled by the use of wavelet transform.
2. **Contourlet Transform:** In painting lines and contours are used instead of dots to create images. The image wavelet representation is equivalent to using points, in this case the image is not clear and y the image elaboration is harder. Likewise, the representation called "contourlets" [20] is the equivalent to using contour lines, simplifying the image construction and giving it a realistic appearance.

According to the results of previous studies [20], an efficient representation of an image should satisfy the following characteristics:

1. **Multi-resolution**: The representation must be a successful approximation from the image, considering low and high resolutions.
2. **Localization**: the basic elements must be localized in both spatial and frequency domains.
3. **Critical Sampling**: the representation should form a basis or a frame with low level of redundancy.
4. **Directionality**: A remarkable representation must have base elements in different directions.
5. **Anisotropy**: To capture smooth contours in images, the representation should contain basis elements using a variety of elongated shapes with different aspect ratios.

The wavelets transforms cover the first three properties, as ridgelets cover the first four, and contourlets cover all of them. After defining the statistical models for ridgelet and contourlet coefficients, the feature extraction is performed. For each subband of a wavelet-based transform, eight statistical features are calculated from the coefficients themselves and the error prediction between the coefficients by using the statistical models proposed. For the final steps, sequential floating search (SFS) method for the feature selection is applied and a SVM [21] for the feature classification is used.

Since the wavelet-based method considers 216 features (useful only for one dimension representation), while the ridgelet-based approach takes into account 48 features, and contourlets approach considers a total of 768 features. The improved results applying both ridgelet and contourlet transforms are reasonable due to the fact that we get the statistics over more than three directions, taking into account all five of the properties of an efficient image representation. The ridgelet and contourlet models are not only effective at separating the different models, but also they separate the images of the two different cameras or scanners with the same model. However, we could try improvements by experimenting with different feature selection algorithms (e. g. SFFP).

3 CONCLUSIONS

In this paper we have studied different existing techniques for solving the image source identification problem. We categorized them into five primary groups according to the processing strategy that they apply: Metadata, Image Features, CFA and Demosaicking Artefacts, Use of Sensor Imperfection and Wavelet Transforms. The main idea of the proposed approaches in each category is described in detail, and reported results are discussed to evaluate the potential of the methods.

Table 1 summarizes the results obtained in the experiments of the different approaches, pointing out the conditions under which they were performed such as: technique, kind of classifier, classifier kernel type, and number of brands, number of models, and number of images, resolutions, and image formats. Outstanding results were found in the analysis: [14] gets closer to reality while considering an open set scenario where usually it is unknown if the images were generated by one of the cameras under investigation, besides defining the ROIs that allow them to work with different resolutions keeping the important information from images. [18] proposing the use of ridgelet and contourlet transform-based image models, taking into account the properties for efficient image representation.

Several experiments have been focused only on traditional cameras leaving out digital camera mobile phones, this deserves special mention owing to the fact that nowadays the number of this kind of devices is increasing rapidly and this trend is expected to continue. Moreover, some experiments do not contemplate different models from the same brand, and those who do it only show results of experiments with one or two models from one brand. It also should be mentioned that databases of images for training and testing are not large enough to represent realistic scenarios.

Through research significant enhancements have been achieved concerning image source identification. Nonetheless, the next steps in the field should be aimed to bridge the aforementioned remaining gaps.

Table 1. Evaluation of camera identification techniques

Group Technique	Based on Image Features			Based on CFA and Demosaicing Artifacts		Based on Sensor Imperfection			Based on Wavelet Transform		
Technique	[7]	[8]	[9]	[1]	[10]	[11]	[12]	[14]	[15]	[17]	[19]
Classifier	SVM	SVM	SVM	SVM	KNN SVM	NA	NA	SVM	SVM	SVM	SVM
SVM Kernel type	Linear	Linear	Linear and Non-linear RBF	Linear and Non-linear RBF	Linear and Non-linear RBF	NA	NA	Non-linear RBF	Non-linear RBF	Non-linear RBF	Non-linear RBF
Number of brands	2	5	4	5	3	1	5	9	4	4	3
Number of models	4	5	5	2	9	2	9	25	6	6	3
Number of Images per camera	150 (60 training 90 testing)	100 (90 training 10 testing)	599	600	200	NA	320	50	350 (100 training 150 testing)	350 (100 training 150 testing)	2000 (1000 training 1000 testing)
Resolutions	1600x 1200	NA	Different	Different	Different	640x 480	Different	Different	Different	Different	NA
Image Format	JPEG	JPEG	JPEG	JPEG	NA	NA	JPEG	JPEG	JPEG	JPEG	NA
Applied to mobiles	Yes	Yes	Yes	No	Yes	No	No	Only 2 among the 25 cameras	No	No	No
Applied to different models from same brand	Yes	No	Yes	Yes	Yes	NA	Yes	Yes	Yes	Yes	No
Average Accuracy (%)	92	97.7	86.36 - 99.91	88 and 84.8	62	NA	NA	94-98	97	98	Wavelet 93.3 Ridgelet 96.7 Contourlet: 99.7

ACKNOWLEDGMENTS

This work was supported by the Agencia Española de Cooperación Internacional para el Desarrollo (AECID, Spain) through Acción Integrada MAEC-AECID MEDITERRÁNEO A1/037528/11. Jocelin Rosales Corripio was also supported by Fundación General UCM through the program AYUDAS DE POSTGRADO SANTANDER - CONVOCATORIA ECL 2012.

References

[1] S. Bayram, H. T. Sencar, and N. Memon. Classification of Digital Camera-Models Based on Demosaicing Artifacts. *Digital Investigation*, 5(1-2), pp. 49–59, 2008.

[2] N. Lloret Romero, V. Gimenez Chornet, J. Serrano Cobos, A. Selles Carot, F. Canet Centellas, and M. Cabrera Mendez. Recovery of Descriptive Information in Images from Digital Libraries by Means of EXIF Metadata. *Library Hi Tech*, 26(2), pp. 302–315, 2008.

[3] J. Boutell, M.and Luo. Photo Classification by Integrating Image Content and Camera Metadata. *In Proceedings of the 17th International Conference on Pattern Recognition*, 4, pp. 901–904. IEEE Computer Society, 2004.

[4] J. Tesic. Metadata Practices for Consumer Photos. *IEEE Multimedia*, 12(3), pp. 86–92, 2005.

[5] M. Boutell and J. Luo. Beyond Pixels: Exploiting Camera Metadata for photo Classification. *Pattern Recognition*, 38(6), pp. 935–946, 2005.

[6] Rick Baer. Resolution Limits in Digital Photography: The Looming End of the Pixel Wars -OSA technical Digest (CD). *In Proceedings of the Imaging Systems*. Optical Society of America, 2010.

[7] M.-J. Tsai, C.-L. Lai, and J. Liu. Camera/Mobile Phone Source Identification for Digital Forensics. *In Proceedings of the IEEE International Conference on Acoustics Speech and Signal Processing*, pages II–221–II–224, 2007.

[8] C. Mckay, A. Swaminathan, H. Gou, and M. Wu. Image Acquisition Forensics: Forensic Analysis to Identify Imaging Source. *In Proceedings of the IEEE International Conference on Acoustics Speech and Signal Processing*, pp.1657– 1660, 2008.

[9] Q. Liu, X. Li, L. Chen, H. Cho, A.P. Cooper, Z. Chen, M. Qiao and A. H. Sung. Identification of Smartphone-Image Source and Manipulation. *Advanced Research in Applied Artificial Intelligence*. LNCS 7345, 2012, pp. 262-271.

[10] O. Celiktutan, I. Avcibas, B. Sankur, N. P. Ayerden, and C. Capar. Source Cell-Phone Identification. *In Proceedings of the IEEE 14th Signal Processing and Communications Applications*, pp. 1–3, 2006.

[11] Z. J. Geradts, J. Bijhold, M. Kieft, K. Kurosawa, K. Kuroki, and N. Saitoh. Methods for Identification of Images Acquired with Digital Cameras. *In Proceedings of the SPIE*, 4232, pp. 505–512. Spie, 2001.

[12] J. Luka, J. Fridrich, and M. Goljan. Digital Camera Identification from Sensor Pattern Noise. *IEEE Transactions on Information Forensics and Security*, 1(2), pp. 205–214, 2006.

[13] T. Van Lanh, K.-S. Chong, S. Emmanuel, and M. S. Kankanhalli. A Survey on Digital Camera Image Forensic Methods. *In Proceedings of the IEEE International Conference on Multimedia and Expo 2007*, pp. 16–19, 2007.

[14] F. D. O. Costa, M. Eckmann, W. J. Scheirer, A. Rocha, Open Set Source Camera Attribution,Graphics. *In Proceedings of the 25th Conference on Patterns and Images*, pp. 71-78, 22-25 August 2012

[15] F. J. Meng, X. W. Kong, and X. G. You. Source Camera Identification Based on Image Bi-Coherence and Wavelet Features. *In Proceedings of the Fourth Annual IFIP WG 11.9 International Conference on Digital Forensics*, Kyoto, Japan, 2008.

[16] P. Pudil, J. Novovicová, and J. Kittler. Floating Search Methods in Feature Selection. *Pattern Recognition Letters*, 15(11), pp. 1119–1125, November 1994.

[17] B. Wang, Y. Guo, X. Kong, and F. Meng. Source Camera Identification Forensics Based on Wavelet Features. *In Proceedings of the International Conference on Intelligent Information Hiding and Multimedia Signal Processing*, pp. 702–705, Los Alamitos, CA, USA, 2009. IEEE Computer Society.

[18] T. Randen and J. H. Husøy. Filtering for Texture Classification: A Comparative Study. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 21(4), pp. 291–310, April 1999.

[19] L. Ozparlak and I. Avcibas. Differentiating Between Images Using Wavelet-Based Transforms: A Comparative Study. *IEEE Transactions on Information Forensics and Security*, 6(4), pp. 1418–1431, December 2011.

[20] M. N. Do and M. Vetterli. The Contourlet Transform: An Efficient Directional Multiresolution Image Representation. *IEEE Transactions on Image Processing*, 14(12), pp. 2091–2106, December 2005.

[21] C. Chang and C. Lin. LIBSVM: A Library for Support Vector Machines. *Technical Report*, 2001.

Day 2

09:00-09:10		Welcome by Chair
09:10-09:40		**Keynote 2: Investigating child abuse images: how technology is closing the net on offenders**, Sharon Girling OBE.
09:40-11:20		**Special Session: EU Projects**
09:40-10:05	EU1	**Dr. Massimo Ciscato** (European Commission): "Overview of main FP7 Security Research Projects and Introducing funding opportunities for the new Horizon 2020 EU Framework Programme".
10:05-10:30	EU2	**Prof. Atta Baddi** (University of Reading, UK): VideoSense European Centre of Excellence in Surveillance VideoAnalytics: *Architectural and Operational Privacy Protection within an Inter-disciplinary Research Framework*
10:30-10:55	EU3	**Mrs. Carmela Occhipinti** (Engineering Ingegneria Informatica S.p.A., Italy): The ADVISE project
10:55-11:20		**Keynote 3: Andrew Rennison,** Surveillance Camera Commissioner
11:20-11:50		**Break, Posters and Exhibition (Room JG2002)** (For the list of posters please see above)
11:50-12:50		**Oral Papers: Vehicles and Crowds**
11:50-12:10	P12	Real-Time Global Anomaly Detection for Crowd Video Surveillance Using SIFT, Puren Guler, Alptekin Temizel, Tugba Taskaya Temizel
12:10-12:30	P13	The Effect of Retro-reflectivity and Reflectance of UK Number Plates on ANPR Performance, Robert Gurney, Michael Rhead, William E Martin, Soodamani Ramalingam, Neil Cohen
12:30-12:50	P14	Vehicle Logo Recognition Using Local Fisher Discriminant Analysis, Simi Wang, Sateesh Pedagadi, James Orwell, Gordon Hunter
12:50-13:45		**Lunch, Posters and Exhibition (Room JG2002)** (For the list of posters please see above)
13:45-14:15		**Keynote 4: Catching Criminals Caught on Camera - How the Met Police is leading the world** Mick Neville, Detective Chief Inspector (DCI), Metropolitan Police Service
14:15-15:15		**Oral Papers: Forensics**
14:15-14:35	P15	Picture-to-Identity linking of social network accounts based on Sensor Pattern Noise, Riccardo Satta, Pasquale Stirparo
14:35:14:55	P16	Source Smartphone Identification Using Sensor Pattern Noise and Wavelet Transform, Jocelin Rosales Corripio, David Manuel Arenas González, Ana Lucila Sandoval Orozco, Luis Javier Garcia Villalba, Julio Hernandez-Castro, Stuart James Gibson
14:55-15:15	P17	Finding Jane Doe: A Forensic Application of 2D Image Calibration, Abby Stylianou, Austin Abrams, Robert Pless
15:15-15:45		**Break, Posters and Exhibition (Room JG2002)** (For the list of posters please see above)
15:45-16:00		**Best Paper/Poster Prizes Ceremony**
16:00-16:30		Panel Discussion
16:30-16:40		Concluding remarks from Chairman

Local organisation:

Kingston University London
Digital Imaging Research Centre

Source Smartphone Identification Using Sensor Pattern Noise and Wavelet Transform

Jocelin Rosales Corripio[1], David Manuel Arenas González[1], Ana Lucila Sandoval Orozco[1], Luis Javier García Villalba[1], Julio Hernandez-Castro[2], Stuart James Gibson[3]

[1] Group of Analysis, Security and Systems (GASS)
Department of Software Engineering and Artificial Intelligence (DISIA)
School of Computer Science, Office 431, Universidad Complutense de Madrid (UCM)
Calle Profesor José García Santesmases s/n, Ciudad Universitaria, 28040 Madrid, Spain
[2] School of Computing, University of Kent, Canterbury CT2 7NF, UK
[3] School of Physical Sciences, University of Kent
Canterbury, Kent, United Kingdom, CT2 7NH

Keywords: Digital Image, Forensics Analysis, Photo Response Non Uniformity, PRNU.

Abstract

The ability to relate a digital photograph to its source camera has application in the areas of digital forensics and multimedia data mining. The majority of previous research in this area has focused on primary function imaging devices (i.e. digital cameras). In this work we use the pattern noise of an imaging sensor to classify digital photographs according to the source smartphone from which they originated. This is timely work as new smartphone models large imaging sensors, afford significant improvements in classification rates using pattern noise. Our approach is to extract wavelet based features which are then classified using a support vector machine. We show that this method generalises well when the number of source cameras is increased.

1 Introduction

Due to increasing storage capacity, usability, portability and affordability, camera enabled mobile phones have become ubiquitous consumer electronic devices. The extensive use of smartphone cameras makes enforcing legal restrictions on the capture and sharing of digital photographs very difficult. Restrictions on the use of cameras include locations such as schools, government offices and businesses. Consequently, tools which permit the identification of source devices have significant utility various areas of law enforcement [2] such as child protection and digital rights management.

2 Source Camera Identification Techniques

Research in this field typically determines make and model by identifying characteristic artefacts within an image. The success of these techniques depend on the assumption that the characteristics are unique to each device [21]. The main problem with this approach is that different models of digital camera are often built using the same core components that originate from a small number of manufacturers. As a consequence it can be difficult, or in some cases impossible, to differentiate between models using such methods.

Numerous approaches to the camera identification have been explored. It has been suggested [8] that the lens radial distortion is the best technique for source identification. Radial distortion causes straight lines to appear as curves in images. The degree of radial distortion for each image can be measured by a process consisting of three steps: edge detection, distorted segment extraction, and distortion error measurement. They experimented with three different cameras and obtained 91.28% source camera identification accuracy.

In [3] an algorithm for identifying and classifying color interpolation operations is presented. The method comprises two algorithms: the first algorithm analyses the correlation of each pixel value with values of its neighbouring pixels, the second analyses the differences between pixels independently. The source camera identification results with images from four to five different models resulted an accuracy of 88% and 84.8% respectively.

Between pixel correlations for source identification were also used in [14], obtaining a coefficient matrix for each color channel while defining a pixel quadratic correlation model. A neural network classifier was used, achieving a success rate of 98.6%. This method is not effective in differentiating between models originating from the same manufacturer.

A set of binary similarity measures is used in [4] as metrics to estimate the similarity between image bit planes. The fundamental assumption of this work is that colour filter array *Color Filter Array* (CFA) demosaicing algorithms, from each make, leave correlations along image bit planes and can be represented by a set of 108 binary similarity measures for classification. The success rate of the experiment was between 81% and 98% when attempting to classify three cameras which

decreased to 62% when nine cameras were considered.

In [17] the authors extend the source identification to different devices such as mobiles phones, digital cameras and scanners. Colour interpolation coefficients and noise characteristics are used for classification. Their experiments showed an overall result of 93.75% accuracy. When identifying the make and model of five mobile phone models, a 97.7% accuracy was obtained.

A method based on bi-coherence statistics phases and magnitudes along with the wavelet coefficients is described in [18]. This method captures the unique nonlinear distortions in the wavelet domain produced by the cameras when performing processing operations over images. An accuracy of 97% in the identification was obtained in distinguishing different models from the same manufacturer.

In [22] a method for identifying the source camera through wavelet feature statistics is presented. The dominant wavelet domain features are extracted to integrate a statistical model of image including 216 first-order wavelet features and 135 co-occurrence second order characteristics. In this study wavelet domain characteristics are considered the most representative and are preferred over the spatial characteristics (color of the image and *Image Quality Metrics* (IQM)) and CFA. Under the same conditions as in the experiments performed in [18] fail to distinguish between different models of the same maker, the average accuracy rate was 98%.

A technique to differentiate images using the wavelet family transforms is described in [19]. Ridgelets and contourlets subband statistical models are proposed to extract the representative features from images. Experiments were conducted to identify three different cameras obtaining accuracies of: 93.3% with wavelet-based approach, 96.7% using ridgelets, and 99.7% with contourlets.

In [13] a method using the marginal density of Discrete Cosine Transformation *Discrete Cosine Transform* (DCT) coefficients in low-frequency coordinates and neighbouring joint density features, on both intra-block and inter-block, from the DCT domain is proposed. In experiments with images of different sizes, from five smartphone models of four manufacturers, an accuracy between 86.36% and 99.91% was obtained.

The techniques based on sensor noise rely on studying the traces left by sensor defects in images. There are broadly two different approaches: pixel defects and sensor pattern noise *Sensor Pattern Noise* (SPN). Pixel defects include hot pixels, dead pixels, the row or column defects and group defects. The SPN method estimates a device 'fingerprint' by averaging multiple residual noise images computed by the application of a denoising filter. The presence of the pattern is determined using a correlation method or machine classification such as support vector machines *Support Vector Machine* (SVM).

In [10], pixel defects of *Charge Coupled Device* (CCD) sensors are studied, focusing on different image features and identify their source. The sources considered were CCD sensor defects, the file format, image noise and watermarking introduced by manufacturer. CCD sensor defects included hot spots, dead pixels, group defects, and row/column defects. Results indicated that each camera has a different defect pattern. Nevertheless, it is also noted that the number of pixel defects for images from the same camera is different and varies greatly depending in the image content. Similarly, it was shown that the number of defects varies with temperature. The study concluded that high quality CCD cameras produce images with fewer defects than other sensor types. When considering only defective CCD sensors, this study is not applicable to the analysis of images generated by mobile devices.

SPN is also used by [15] to create fingerprints which were used to uniquely identify cameras of different make and model. To identify the camera from a given image, the reference pattern is considered as a watermark in the image and its presence is established by a correlation detector. It was found that this method is affected by processing algorithms such as image JPEG compression and gamma correction. The results for pictures with different sizes were unsatisfactory [21].

In [9] an approach to source camera identification using an 'open set' scenario is proposed for which, unlike previous work, the access to the source camera is not required to perform the analysis. This approach, in contrast to others, considers 9 different regions of interest *Region Of Interest*s (ROIs) located in the corners and the center of the images (not only the central region of the image). Using these ROIs it is possible to work with different resolution images without requiring zero padding or color interpolation. The SPN is computed for each color channel generating a total of 36 representative features for each image. Then, the image features are labelled as positive class (created from particular camera) or negative class (originating from another camera). After the SVM training phase, the separating hyperplane is moved by a given amount either inward (for positive classes) or out (for negative classes) for to accommodate the open set scenario. The results of their experiments had an accuracy of 94.49%, 96.77% and 98.10%.

The basic SPN method described in [15] is developed further by [11]. They propose that a stronger component of the sensor noise is less reliable and therefore it should be attenuated. They performed experiments with six different cameras. For images of 1536×2048 pixels, they obtained an accuracy of 38.5 % with the implementation without the improvement and 80.8% with the proposed improvement. For images of 512×512 pixels they obtained an accuracy of 21.8% without improvement and 78.7% with the proposed improvement.

A detailed comparison of different source identification techniques is presented in [20].

3 Source Identification Algorithm

Previous work has shown sensor pattern noise [10] [11] [15] and wavelet transform [18, 19] to be an effective method for source camera identification. However, almost all studies have focused only traditional cameras excluding mobile cameras. This makes it an area of study that requires attention. Using a biometric analogy, we consider each noise pattern to be a fingerprint of its source camera's sensor. In our study, sensor pattern noise is used to classify images captured by, camera enabled, smartphones. Our approach characterises the fingerprints using wavelet based feature vectors. The scheme pre-

sented in Figure 1 shows the functional diagram of our proposal.

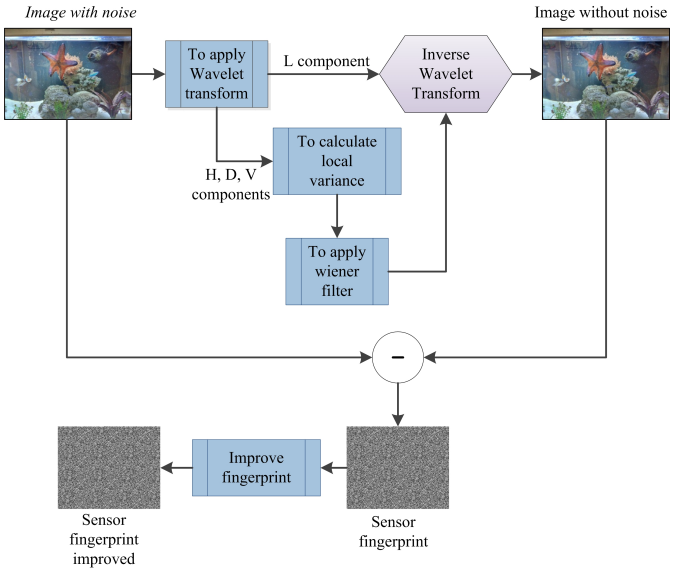


Figure 1. Scheme functional.

Noise images were obtained using the method previously described by [15] also summarised by Algorithm 1 as follows.

Algorithm 1: Extracting PRNU

Input: Image I
Variance estimation: adaptive or non-adaptive
Result: Sensor fingerprint I_{noise}

1. **procedure** EXTRACTPRNU(I)
2. Apply a wavelet decomposition in 4 levels to I;
3. **foreach** *wavelet decomposition level* **do**
4. **foreach** *component* $c \in \{H, V, D\}$ **do**
5. Compute the local variance;
6. **if** *adaptive variance* **then**
7. Compute 4 variances with windows of size: 3, 5, 7 and 9 respectively;
8. Select the minimum variance;
   else
9. Compute the variance with a window of size 3;
10. Compute noiseless wavelet components applying the Wiener filter to the variance;
11. Obtain I_{clean} by applying the inverse wavelet transform with clean components calculated;
12. Obtain the sensor noise with $I_{noise}=I-I_{clean}$;
13. Apply zero-meaning to I_{noise};
14. Increase the green channel weight with $I_{noise}= \quad 0.3\cdot I_{noise_R}+0.6\cdot I_{noise_G}+0.1\cdot I_{noise_B}$;
15. **end procedure**

To extract its noise pattern, an image is decomposed into its red, green and blue color channels. Then, a four-level wavelet decomposition of each color channel is calculated using the Daubechies, 8-tap, *Separable Quadrate Mirror Filters* (QMF). The number of decomposition levels can be increased to improve accuracy or reduced to reduce processing time.

Horizonal H, vertical V and diagonal D high-frequency images are obtained for each level of decomposition. For each detail image, the local scene variance in a $W \times W$ window is estimated. Four estimates are obtained with window sizes corresponding to $W \in \{3, 5, 7, 9\}$. Finally, we choose the estimate which maximises the a-posteriori probability (MAP).

$$\hat{\sigma}^2(i,j)=\max\left(0, \frac{1}{W^2}\sum_{(i,j)\in N} c^2(i,j) - \sigma_0^2\right), \quad (i,j)\in J \qquad (1)$$

Where, $c(i,j)$ is the high-frequency component and $c \in \{H, V, D\}$; σ_0 controls the degree of noise suppression.

The minimum of four variances is chosen as the best estimate:

$$\hat{\sigma}^2(i,j)=\min\left(\sigma_3^2(i,j),\sigma_5^2(i,j),\sigma_7^2(i,j),\sigma_9^2(i,j)\right), \quad (i,j)\in J \qquad (2)$$

An alternative, and less accurate method, is to simply use $W = 3$ as the estimated local variance.

The denoised wavelet coefficients are defined by the Wiener filter as follows:

$$c_{clean}(i,j) = \quad c(i,j)\frac{\hat{\sigma}^2(i,j)}{\hat{\sigma}^2(i,j)+\sigma_0^2} \qquad (3)$$

The noise residual is obtained by calculating the inverse transform and subtracting the denoised image from the original image. JPEG and demosaicing artefacts, present in the noise image, are suppressed by subtracting the mean column and row values [7]. Greater weight is given to the green channel since due to the configuration of the color matrix this channel contains more information about the image [5, 16, 1].

The next step is to obtain features that characterise the sensor fingerprint for the purpose of classification. A total of 81 features (3 channels×3 wavelet components×9 central moments) is extracted using the Algorithm 2 as follows:

Algorithm 2: Extracting features

Input: Sensor fingerprint I_{noise}
Result: 81 features

1. **procedure** EXTRACTFEATURES(I)
2. Separate R, G and B color channels of I_{noise};
3. **foreach** *color channel* **do**
4. Apply a wavelet decomposition in 1 level;
5. **foreach** *component* $c \in \{H, V, D\}$ **do**
6. Compute k central moments with $m_k = \frac{1}{n}\sum_{i=1}^{n} |c_i - \overline{c}|^k$;
7. **end procedure**

Classification was performed using a SVM of the RBF kernel. We used the LibSVM package in which the SVM is extended to multiple classes yielding class probability estimates [6]. The kernel parameter $\gamma = 2^3$ and cost parameter $C = 32768$ were used for the SVM. A grid search was used to obtain the best kernel parameters (γ and C). The classifier was trained and tested with feature vectors extracted from randomly selected images.

4 Experiments and Results

To assess the effectiveness of the proposed algorithms, two experiments were conducted considering the central 1024x1024 pixel image block, as recommended in [12]. Table 1 summarises the experimental conditions used in our algorithms.

Table 1. Parameters used in the proposed algorithms

Parameter	Value
Dimensions	1024 x1024
Number of training photos by camera	100
Number of testing photos by camera	100
Variance estimation	Non-adaptive

The mobile device digital cameras used and their configurations are showed in Table 2.

Table 2. Configurations used in mobile device digital cameras

Brand	Model	Resolution	Taking Conditions
Apple	iPhone3G (A1)	2 MP (1600x1200)	Scene type: Any
	iPhone4S (A2)	8 MP (3264x2448)	Orientation: Vertical
	iPhone3 (A3)	2 MP (1600x1200)	Flash: Disabled
	iPhone5 (A4)	8 MP (3264x2448)	Light: Natural
Black Berry	8520 (B1)	2 MP (1600x1200)	White balance: Auto
Sony Ericsson	UST25a (SE1)	5 MP (2592x1944)	Digital zoom ratio: 0
	U5I (SE2)	8 MP (3264x2448)	Exposure time: 0 seg
Samsung	GT-I9100 (S1)	8 MP (3264x2448)	ISO speed: Automatic
	GT-S5830 (S2)	5 MP (2592x1944)	
	GT-S5830M (S2)	5 MP (2592x1944)	
LG	E400 (L1)	3.2 MP (2048x1536)	
HTC	DesireHD (H1)	8 MP (3264x2448)	
Nokia	E61I (N1)	2 MP (1600x1200)	

4.1 Experiment 1

In this experiment, a group of 8 mobile device digital cameras from 4 different manufacturers was tested. From Apple, the models iPhone3G (A1), iPhone4S (A2), and iPhone3 (A3) were considered; from BlackBerry the 8520 (B1); from Sony Ericsson the UST25a (SE1) and the U5I (SE2); and from Samsung the GTI9100 (S1) and the GTS5830 (S2) models.

The performance of the classifier was tested 10 times, using a 10 different random samples of 100 images, and the average classification rate recorded. The performance changed only slightly in each run which indicates stability over different training and testing image sets.

The PRNU extraction algorithm and feature extraction algorithm are implemented in Python 2.7 with an Intel Core i5, 2.5-GHz processor and 8 GB of RAM. It takes approximately 40s to extract the PRNU and compute the features for a single image. Training the SVM classifier and testing is realized in a 2s and fraction of a second respectively. A random sample of 100 images was used for testing a different random sample of 100 images was used for testing.

Sample confusion tables from eight camera groups are given below. The best, middle, worst case tables are show in Tables 3, 4 and 5 respectively. The average accuracy for correctly identifying camera make and model was 93.2%.

Table 3. Confusion matrix of best result (93.87%)

Camera	A1	A2	A3	B1	SE1	SE2	S1	S2
A1	**96**	1	0	0	0	0	0	3
A2	0	**97**	0	0	0	0	3	0
A3	0	0	**98**	0	0	0	2	0
B1	0	0	0	**94**	0	4	0	2
SE1	11	1	0	0	**88**	0	0	0
SE2	3	0	0	1	0	**93**	1	2
S1	4	8	0	0	0	3	**85**	0
S2	0	0	0	0	0	0	0	**100**

Table 4. Confusion matrix of middle result (93.25%)

Camera	A1	A2	A3	B1	SE1	SE2	S1	S2
A1	**94**	1	0	0	0	1	0	4
A2	0	**96**	0	0	1	0	3	0
A3	0	0	**97**	0	0	0	2	1
B1	0	0	0	**94**	0	2	0	4
SE1	10	1	0	0	**89**	0	0	0
SE2	2	0	0	1	0	**94**	1	2
S1	5	6	0	0	0	6	**83**	0
S2	0	0	0	0	0	1	0	**99**

Table 5. Confusion matrix of worst result (92.62%)

Camera	A1	A2	A3	B1	SE1	SE2	S1	S2
A1	**92**	1	0	0	0	0	0	7
A2	0	**96**	0	0	1	0	3	0
A3	0	1	**99**	0	0	0	0	0
B1	0	0	3	**91**	0	4	0	2
SE1	7	2	0	0	**91**	0	0	0
SE2	2	0	0	1	0	**94**	1	2
S1	4	10	0	0	0	7	**79**	0
S2	0	0	0	0	0	1	0	**99**

4.2 Experiment 2

In order to evaluate the scalability of the method to a larger number of classes, a group of 14 mobile device digital cameras from 7 different manufacturers was used. From Apple,

Table 6. Confusion matrix of experiment 2

Camera	A1	A2	A3	A4	B1	SE1	SE1	S1	S1	S3	L1	H1	H2	N1
A1	**90**	0	0	2	0	0	0	0	7	0	1	0	0	0
A2	0	**91**	0	3	0	0	0	3	0	0	0	1	2	0
A3	0	0	**98**	0	0	0	0	2	0	0	0	0	0	0
A4	0	0	1	**88**	0	0	0	0	0	0	3	6	0	2
B1	0	0	0	2	**73**	0	0	0	4	0	0	1	0	20
SE1	7	0	0	0	0	**80**	0	0	0	0	1	12	0	0
SE2	1	0	0	2	2	0	**86**	1	2	5	1	0	0	0
S1	4	5	0	4	0	0	1	**83**	0	0	1	0	2	0
S2	0	0	0	0	0	0	0	0	**100**	0	0	0	0	0
S3	0	0	1	0	0	0	8	0	0	**85**	0	1	0	5
L1	0	0	0	9	0	6	0	0	2	0	**70**	13	0	0
H1	2	0	0	0	0	11	0	0	1	0	1	**85**	0	0
H2	0	6	0	0	0	0	0	0	0	0	0	0	**94**	0
N1	0	0	0	0	2	0	0	0	0	0	0	0	0	**98**

the models iPhone3G (A1), iPhone4S (A2), iPhone3 (A3) and iPhone5 (A4) were considered; from BlackBerry the 8520 (B1); from Sony Ericsson the UST25a (SE1) and the U5I (SE2); from Samsung the GTI9100 (S1), the GTS5830 (S2) and the GT-S5830M (S3); from Lg the E400 (L1); from HTC the DesireHD (H1) and the Desire (H2); finally from Nokia the E61I (N1) model. The average classification rate dropped to 87.214% as shown in the confusion matrix of Table 6 indicating a small loss in performance when the number of classes (cameras) is increased.

5 Conclusion

A method for source camera identification, based wavelet features of image noise residuals and SVM classification, was tested on photographs acquired from a range of smartphones. In the first experiment 8 models from 4 manufacturers were considered resulting in an overall accuracy of 93.2%. In order to evaluate the scalability of the approach, we repeated the experiment using 14 models from 7 manufactures and achieved an average success rate of 87.214%. Our results, tentatively, suggest that the method is applicable to data sets containing images from a large number of different cameras and therefore the method promises potential utility for digital forensics and data mining applications.

Acknowledgements

Part of the computations of this work were performed in EOLO, the HPC of Climate Change of the International Campus of Excellence of Moncloa, funded by MECD and MICINN. This is a contribution to CEI Moncloa.

References

[1] J. Adams, K. Parulski, and K. Spaulding. Color Processing in Digital Cameras. *Micro, IEEE*, 18(6):20–30, December 1998.

[2] M. Al-Zarouni. Mobile Handset Forensic Evidence: a Challenge for Law Enforcement. In *Proceedings of the 4th Australian Digital Forensics Conference*. School of Computer and Information Science, Edith Cowan University, December 2006.

[3] S. Bayram, H. T. Sencar, and N. Memon. Classification of Digital Camera-Models Based on Demosaicing Artifacts. *Digital Investigation*, 5(1-2):49–59, September 2008.

[4] O. Celiktutan, I. Avcibas, B. Sankur, N. P. Ayerden, and C. Capar. Source Cell-Phone Identification. In *Proceedings of the IEEE 14th Signal Processing and Communications Applications*, pages 1–3. IEEE, April 2006.

[5] O. Celiktutan, B. Sankur, and I. Avcibas. Blind Identification of Source Cell-Phone Model. *IEEE Transactions on Information Forensics and Security*, 3(3):553–566, September 2008.

[6] C. C. Chang and C. J. Lin. LIBSVM: A Library for Support Vector Machines. Version 3.17, Abril 26, 2013, http://www.csie.ntu.edu.tw/cjlin/libsvm/.

[7] M. Chen, J. Fridrich, M. Goljan, and J. Lukas. Determining Image Origin and Integrity Using Sensor Noise. *IEEE Transactions on Information Forensics and Security*, 3(1):74–90, March 2008.

[8] K. S. Choi. Source Camera Identification Using Footprints From Lens Aberration. In *Proceedings on Digital Photography II*, number 852 in 6069, pages 60690J–60690J–8. SPIE International Society For Optical Engineering, February 2006.

[9] F. D. O. Costa, M. Eckmann, W. J. Scheirer, and A. Rocha. Open Set Source Camera Attribution. In *Proceedings of the 25th Conference on Graphics, Patterns and Images*, pages 71–78. IEEE, August 2012.

[10] Z. J. Geradts, J. Bijhold, M. Kieft, K. Kurosawa, K. Kuroki, and N. Saitoh. Methods for Identification of Images Acquired with Digital Cameras. In *Proceedings on Enabling Technologies for Law Enforcement and Security*, volume 4232, pages 505–512. SPIE-International Society for Optical Engine, February 2001.

[11] C. T. Li. Source Camera Linking Using eEnhanced Sensor Pattern Noise Extracted from Images. In *3rd International Conference on Crime Detection and Prevention (ICDP 2009)*, pages 1–6. Curran Associates, Inc., December 2009.

[12] C. T. Li and R. Satta. On the Location-Dependent Quality of the Sensor Pattern Noise and its Implication in Multimedia Forensics. In *4th International Conference on Imaging for Crime Detection and Prevention 2011 (ICDP 2011)*, pages 1–6. Curran Associates, Inc., November 2011.

[13] Q. Liu, X. Li, L. Chen, H. Cho, A. P. Cooper, Z. Chen, M. Qiao, and A. H. Sung. Identification of Smartphone-Image Source and Manipulation. In He Jiang, Wei Ding, Moonis Ali, and Xindong Wu, editors, *Advanced Research in Applied Artificial Intelligence*, volume 7345 of *Lecture Notes in Computer Science*, pages 262–271. Springer Berlin Heidelberg, Dalian, China, June 2012.

[14] Y. Long and Y. Huang. Image Based Source Camera Identification using Demosaicking. In *Proceedings of the IEEE 8th Workshop on Multimedia Signal Processing*, pages 419–424. IEEE, October 2006.

[15] J. Lukas, J. Fridrich, and M. Goljan. Digital Camera Identification from Sensor Pattern Noise. *IEEE Transactions on Information Forensics and Security*, 1(2):205–214, June 2006.

[16] C. McKay. Forensic Analysis of Digital Imaging Devices. Technical report, University of Maryland, 2007.

[17] C. Mckay, A. Swaminathan, H. Gou, and M. Wu. Image Acquisition Forensics: Forensic Analysis to Identify Imaging Source. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 1657–1660. IEEE, June 2008.

[18] F. J. Meng, X. W. Kong, and X. G. You. Source Camera Identification Based on Image Bi-Coherence and Wavelet Features. In *Proceedings of the Fourth Annual IFIP WG 11.9 International Conference on Digital Forensics*, Kyoto, Japan, January 2008.

[19] L. Ozparlak and I. Avcibas. Differentiating Between Images Using Wavelet-Based Transforms: A Comparative Study. *IEEE Transactions on Information Forensics and Security*, 6(4):1418–1431, December 2011.

[20] A.L. Sandoval Orozco, D.M. Arenas González, J. Rosales Corripio, L.J. García Villalba, and J.C. Hernandez-Castro. Techniques for Source Camera Identification. In *Proceedings of the 6th International Conference on Information Technology*, pages 1–9, May 2013.

[21] T. Van Lanh, K. S. Chong, S. Emmanuel, and M. S. Kankanhalli. A Survey on Digital Camera Image Forensic Methods. In *Proceedings of the IEEE International Conference on Multimedia and Expo*, pages 16–19. IEEE, July 2007.

[22] B. Wang, Y. Guo, X. Kong, and F. Meng. Source Camera Identification Forensics Based on Wavelet Features. In *Proceedings of the International Conference on Intelligent Information Hiding and Multimedia Signal Processing*, volume 0, pages 702–705. IEEE Computer Society, September 2009.

Rafael Álvarez · Joan Josep Climent · Francisco Ferrández · Francisco M. Martínez
Leandro Tortosa · José Francisco Vicent · Antonio Zamora
(editores)

Actas de la XIII Reunión Española sobre Criptología y Seguridad de la Información

RECSI XIII

Alicante, 2-5 de septiembre de 2014

PUBLICACIONES UNIVERSIDAD DE ALICANTE

Actas de la XIII Reunión Española sobre Criptología y Seguridad de la Información

RECSI XIII

Alicante, 2-5 de septiembre de 2014

Rafael Álvarez · Joan Josep Climent · Francisco Ferrández · Francisco M. Martínez
Leandro Tortosa · José Francisco Vicent · Antonio Zamora
(editores)

Publicaciones de la Universidad de Alicante

Campus de San Vicente, s/n
03690 San Vicente del Raspeig
Publicaciones@ua.es - http://publicaciones.ua.es
Teléfono: 965 903 480

ISBN: 978-84-9717-323-0

Universitat d'Alacant
Universidad de Alicante

Identificación de la Fuente de Imágenes de Dispositivos Móviles Basada en el Ruido del Sensor

Jocelin Rosales Corripio, David Manuel Arenas González, Ana Lucila Sandoval Orozco,
Luis Javier García Villalba

Grupo de Análisis, Seguridad y Sistemas (GASS), Departamento de Ingeniería del Software e Inteligencia Artificial
Facultad de Informática, Despacho 431, Universidad Complutense de Madrid (UCM)
Calle Profesor José García Santesmases, 9, Ciudad Universitaria, 28040 Madrid
Email: jocerosa@ucm.es, {darenas, asandoval, javiergv}@fdi.ucm.es

***Resumen*—La fuente de una imagen digital se puede identificar a través de los rasgos que el dispositivo que la genera impregna en ella durante el proceso de su generación. La mayoría de las investigaciones realizadas en los últimos años sobre técnicas de identificación de fuente se han enfocado únicamente en la identificación de cámaras tradicionales DSC (*Digital Still Camera*). Considerando que hoy en día las cámaras de los dispositivos móviles prácticamente han sustituido a las DSCs se detectó la necesidad de realizar investigación sobre las técnicas para identificar la fuente de imágenes generadas por dispositivos móviles. Las imágenes digitales generadas por una cámara digital contienen intrínsecamente un patrón del ruido del sensor que se puede usar como medio de identificación de la fuente. Específicamente, las cámaras digitales de dispositivos móviles cuentan en su mayoría con un tipo de sensor que deja rasgos característicos en la imagen. En este trabajo se propone un algoritmo basado en el ruido del sensor y en la transformada wavelet para identificar el dispositivo móvil (marca y modelo) que ha generado determinadas imágenes bajo investigación.**

Palabras clave*—Análisis forense, imagen digital, patrón de ruido del sensor, PRNU. *(Forensics analysis, digital image, sensor pattern noise, PRNU).

I. Introduction

Con frecuencia las fotografías son consideradas como una parte de la verdad al ser hechos reales capturados por dispositivos electrónicos (cámaras). Sin embargo, con el desarrollo de la tecnología han surgido herramientas potentes y sofisticadas que facilitan de una manera impresionante la alteración de las imágenes digitales, incluso para quienes no tienen conocimientos técnicos o especializados en el área [1].

El desarrollo de las tecnologías digitales ha estado y continúa avanzando a un ritmo imparable. Cada día el número de cámaras digitales va creciendo, así como la facilidad de acceso a ellas. Las cámaras digitales de móviles merecen especial atención, ya que estudios realizados indican que al final del año 2012 el número total de dispositivos móviles activos alcanzó los 6,7 billones y se estima que para el verano del 2013 este número igualará al total de la población del planeta 7,1 billones. El 83 % de estos dispositivos móviles cuentan con cámara digital integrada, las cuales a diferencia de las cámaras digitales convencionales son llevadas por sus dueños todo el tiempo a la mayoría de lugares que asiste y en muchos casos tienen conexión a internet [2].

Debido al incremento en sus capacidades de almacenamiento, procesamiento, usabilidad y portabilidad así como a su bajo coste, los dispositivos móviles están presentes en diversidad de actividades, lugares y eventos de la vida diaria. A causa del extenso uso de las cámaras digitales de dispositivos móviles se han generado polémicas, discusiones y normas sobre la prohibición de su uso en lugares como escuelas, oficinas de gobierno, eventos empresariales, conciertos, empresas, etc. Una consecuencia más de su extenso uso es que las imágenes digitales en la actualidad son utilizadas como testigos silenciosos en procesos judiciales, siendo una pieza crucial de la evidencia del crimen [3]. Es por ello que contar con herramientas que permitan identificar a los dispositivos que han generado una cierta imagen digital cobra importancia ya que podría servir en diversas áreas como la lucha contra la pornografía infantil, la prevención de robo de tarjetas de crédito, el combate a la piratería, la prevención de secuestros, etc.

II. Técnicas de Análisis Forense en Imagenes

La investigación en este campo estudia el diseño de técnicas para identificar las características, especialmente marca y modelo, de los dispositivos utilizados para la generación de imágenes digitales. El éxito de estas técnicas depende del supuesto de que todas las imágenes adquiridas por un mismo dispositivo presentan características intrínsecas del dispositivo. Las características que se usan para identificar marca y modelo de las cámaras digitales se derivan de las diferencias que existen entre las técnicas de procesamiento de las imágenes y las tecnologías de los componentes que se utilizan [4]. El mayor problema con este enfoque es que los diferentes modelos de las cámaras digitales usan componentes de un número reducido de fabricantes, y que los algoritmos que usan también son muy similares entre modelos de la misma marca. Es por ello que la fiabilidad de la identificación de la cámara fuente depende en gran parte de la identificación de varias características independientes del modelo. Según [4] se pueden establecer cuatro grupos de técnicas para este fin: utilización de la aberración de las lentes, interpolación de la matriz CFA, uso de las características de la imagen e imperfecciones del sensor. Esta última constituye el objeto de

este trabajo. Además de las anteriores existe otro grupo de técnicas basadas en los metadatos.

Las técnicas basadas en el estudio de las huellas que los defectos del sensor dejan sobre las imágenes se dividen en dos ramas: defectos de píxel y patrón de ruido del sensor SPN (Sensor Pattern Noise). En la primera se estudian los defectos de píxel, los píxeles calientes, los píxeles muertos, los defectos de fila o columna, y los defectos de grupo. En la segunda se construye un patrón del ruido promediando los múltiples residuos de ruido obtenidos mediante algún filtro de eliminación de ruido. La presencia del patrón se determina utilizando algún método de clasificación como correlación o máquinas SVM.

En [5] se estudian los defectos de los píxeles en los sensores de tipo CCD, centrándose en la evaluación de diferentes características para examinar las imágenes e identificar la fuente: defectos del sensor CCD, formato de los archivos usados, ruido introducido en la imagen y marcas de agua introducidas por el fabricante de la cámara. Entre los defectos del sensor CCD considerados se encuentran los puntos calientes, los píxeles muertos, los defectos en grupo y los defectos de fila o columna. En sus resultados se observa que cada una de las cámaras tiene un patrón de defecto diferente. Sin embargo, también se señala que el número de defectos en los píxeles para una cámara es diferente entre fotos y varía demasiado en función del contenido de la imagen. Asimismo, se revela que el número de defectos cambia con la temperatura. Al considerar únicamente los defectos de los sensores de tipo CCD este estudio no es aplicable al análisis de imágenes generadas por dispositivos móviles.

En [6] se analiza el patrón de ruido del sensor de un conjunto de cámaras, el cual funciona como una huella dactilar, permitiendo la identificación única de cada cámara. Para obtener este patrón se realiza un promedio del ruido obtenido a partir de diferentes imágenes utilizando un filtro de eliminación de ruido. Para identificar la cámara a partir de una imagen dada, se considera el patrón de referencia como una marca de agua cuya presencia en la imagen es establecida mediante un detector de correlación. El estudio se realizó con 320 imágenes procedentes de 9 modelos distintos de cámaras. También se demuestra que este método está afectado por algoritmos de procesamiento de la imagen como la compresión JPEG y la corrección gamma. Los resultados para fotografías con diferentes tamaños y recortadas no son satisfactorios [4].

En [7] se propone un enfoque para la identificación de la cámara fuente considerando escenarios abiertos, donde a diferencia de los escenarios cerrados no se da por sentado contar con acceso a todas las posibles cámaras de origen de la imagen. Este enfoque, considera 9 diferentes áreas de interés ROI (*Region Of Interest*) que se encuentran en las esquinas y el centro de las imágenes. El uso de las regiones de interés permite trabajar con imágenes de diferentes resoluciones sin la necesidad de rellenar con ceros las imágenes y sin el uso de artefactos de interpolación de color. Para determinar las características se calcula el SPN para cada uno de los canales R, G y B. Asimismo, se calcula el SPN para el canal Y (luminancia), generándose un total de 36 características para representar cada imagen. Después, las imágenes tomadas por la cámara bajo investigación son etiquetadas como la clase positiva y las tomadas por las cámaras disponibles restantes como las clases negativas. Después de la fase de entrenamiento de la SVM en la que se calcula el hiper-plano que separa los casos positivos y negativos toman en cuenta las clases desconocidas del escenario abierto moviendo el hiper-plano generado por un valor dado ya sea hacia adentro (hacia las clases positivas) o hacia afuera (las clases negativas). En los experimentos utilizan un conjunto de 25 cámaras digitales de 9 fabricantes, 150 imágenes en formato JPEG de cada cámara con diferentes configuraciones de luz, zoom y flash. Los resultados de los experimentos mostraron una precisión del 94,49 %, del 96,77 % y del 98,10 %, utilizando conjuntos abiertos con 2/25, 5/25, y 15/25 cámaras, respectivamente, definiendo un conjunto abierto x/y como el conjunto de y cámaras donde x cámaras son usadas para entrenar y probar las imágenes que pueden pertenecer a cualquiera de las cámaras x conocidas, así como a las otras y-x cámaras desconocidas.

En [8] se basan en el trabajo de [6] para extraer el ruido del sensor usando el cálculo de similitudes como método de la clasificación. Exponen que el ruido del sensor puede estar muy contaminado por los detalles de los escenarios y proponen que entre más fuerte es un componente del ruido del sensor es menos fiable y por lo tanto debe ser atenuado. Proponen una forma de atenuar los valores altos del ruido del sensor y realizan experimentos de identificación con 6 cámaras tradicionales diferentes (100 imágenes de cada cámara). Para las imágenes de 1536x2048 pixeles obtuvieron una tasa de acierto del 38.5 % con la implementación sin la mejora y del 80.8 % con la mejora propuesta; para las imágenes de 512x512 píxeles obtuvieron una tasa de acierto del 21.8 % sin la mejora y del 78.7 % con la mejora propuesta.

III. Algoritmo de Identificación de la Fuente

Debido a la propiedad determinista del patrón de ruido del sensor que está presente en cada imagen capturada, se puede usar este patrón como huella para identificar el dispositivo que generó la imagen objeto en investigación. Haciendo una analogía, se puede decir que el patrón del ruido del sensor es para una cámara digital lo que la huella para un ser humano.

Para poder identificar la marca y el modelo de la cámara digital de un dispositivo móvil se requiere de un algoritmo que nos permita extraer el ruido del sensor y otro que nos permita obtener las características de las huellas obtenidas para así poder clasificarlas e identificarlas.

Tomando como referencia las ideas principales de [6] se propone un algoritmo para extraer el ruido del sensor (también conocido como ruido residual) que se describe en el algoritmo 1.

Con el promediado a cero se limpia la huella de las características que no son intrínsecas al sensor aplicando como se sugiere en [9], de tal manera que los promedios de las filas y de las columnas sean iguales a cero. Esto se logra restando el promedio de la columna a cada píxel de la columna y

Algoritmo 1: Extraer ruido del sensor

Input: Imagen
varianza: (adaptativa o no adaptativa)
Result: Huella del sensor de la imagen

1 **procedure** EXTRAERHUELLA(I)
2 Realizar descomposición wavelet de 4 niveles de I_n;
3 **foreach** *nivel de la descomposición wavelet* **do**
4 **foreach** $c \in \{H,V,D\}$ **do**
5 Calcular la varianza local;
6 **if** *varianza adaptativa* **then**
7 Calcular 4 varianzas con ventanas de tamaños 3, 5 ,7 y 9 respectivamente;
8 Seleccionar la varianzas mínima;
else
9 Calcular la varianza con una ventana de tamaño 3;
10 Calcular los componentes wavelet sin ruido aplicando el filtro de Wiener a la varianza;
11 Obtener la imagen limpia del ruido del sensor aplicando la Transformada Inversa Wavelet;
12 Calcular el ruido del sensor con $I_{ruido}=I_{entrada}-I_{limpia}$;
13 Aplicar a I_{ruido} un promediado a cero;
14 Aumentar en I_{ruido} el peso del canal verde con $I_{ruido}=0{,}3\cdot I_{ruido_R}+0{,}6\cdot I_{ruido_G}+0{,}1\cdot I_{ruido_B}$;
15 **end procedure**

Algoritmo 2: Extracción de Características

Input: Imagen
Huella del sensor de la imagen
Result: 81 características

1 **procedure** EXTRAERCARACTERISTICAS(I)
2 Separar los canales R, G y B de la huella del sensor;
3 **foreach** *canal de color* **do**
4 Hacer una descomposición wavelet de un nivel;
5 **foreach** $c \in \{H,V,D\}$ **do**
6 Calcular k momentos centrales con $m_k = \frac{1}{n}\sum_{i=1}^{n}|c_i - \bar{c}|^k$;
7 **end procedure**

posteriormente restando el promedio de la fila a cada píxel de la fila. Esta operación se aplica a todas las filas y columnas de la imagen. Después de limpiar la imagen se le da un mayor peso al canal verde ya que debido a la configuración de la matriz de color éste contiene más información sobre la imagen que el resto de los canales de color [10][11]. La identificación de las cámaras se realiza utilizando una máquina de soporte vectorial SVM para lo que es necesario extraer una serie de características que representen a las huellas de los sensores. Se calculan un total de 81 características (3 canales x 3 componentes wavelet x 9 momentos centrales) mediante el algoritmo 2.

Con las características que se extraen tanto de las imágenes para entrenamiento como para probar se alimenta la máquina SVM y se obtienen las clasificaciones.

IV. EXPERIMENTOS Y RESULTADOS

Para evaluar la efectividad del algoritmo de identificación de la fuente de dispositivos móviles se realizaron dos experimentos, en los que se consideraron los 1024x1024 píxeles centrales de las fotografías como se recomienda ampliamente en [12]. La Tabla I resume los principales parámetros utilizados.

En el primer experimento se probó con un grupo de 8 cámaras digitales de dispositivos móviles de 4 fabricantes. De Apple se consideraron los modelos iPhone3G (A1), iPhone4S (A2) y iPhone3 (A3); de BlackBerry el 8520 (B1); de Sony Ericsson el UST25a (SE1) y el U5I (SE2); y de Samsung el GTI9100 (S1) y el GTS5830 (S2). El algoritmo propuesto obtuvo un porcentaje de acierto promedio de 93.625 % al identificar entre marca y modelo como se observa en la matriz de confusión de la Tabla II.

Tabla I
PARÁMETROS UTILIZADOS EN LOS EXPERIMENTOS

Parámetro	Valor
Tipo de Fotos	Sin ninguna restricción
Dimensiones	1024 x1024
Fotos Entrenadas x Cámara	100
Fotos Probadas x Cámara	100
Cálculo de la Varianza	Enfoque no adaptativo

Tabla II
MATRIZ DE CONFUSIÓN DEL EXPERIMENTO 1

Cámara	A1	A2	A3	B1	SE1	SE2	S1	S2
A1	**92**	1	0	0	0	1	0	6
A2	0	**96**	0	0	1	0	3	0
A3	0	0	**99**	0	0	0	1	0
B1	0	0	0	**94**	0	2	0	4
SE1	7	2	0	0	**91**	0	0	0
SE2	2	0	0	1	0	**94**	1	2
S1	4	8	0	0	0	5	**83**	0
S2	0	0	0	0	0	0	0	**100**

Con la finalidad de acercarse a escenarios más reales el segundo experimento se realizó con 14 cámaras digitales de dispositivos móviles de 7 fabricantes. De Apple se consideraron los modelos iPhone3G (A1), iPhone4S (A2), iPhone3 (A3) y iPhone5 (A4); de BlackBerry el 8520 (B1); de Sony Ericsson el UST25a (SE1) y el U5I (SE2); de Samsung el GTI9100 (S1), el GTS5830 (S2) y el GT-S5830M (S3); de Lg el E400 (L1); de HTC el DesireHD (H1) y el Desire (H2); y de Nokia el E61I (N1). El algoritmo propuesto obtuvo un porcentaje de acierto promedio de 87,214 % como se puede observar en la matriz de confusión de la Tabla III.

Tabla III
MATRIZ DE CONFUSIÓN DEL EXPERIMENTO 2

Cámara	A1	A2	A3	A4	B1	SE1	SE1	S1	S1	S3	L1	H1	H2	N1
A1	**90**	0	0	2	0	0	0	0	7	0	1	0	0	0
A2	0	**91**	0	3	0	0	0	3	0	0	0	1	2	0
A3	0	0	**98**	0	0	0	0	2	0	0	0	0	0	0
A4	0	0	1	**88**	0	0	0	0	0	0	3	6	0	2
B1	0	0	0	2	**73**	0	0	0	4	0	0	1	0	20
SE1	7	0	0	0	0	**80**	0	0	0	0	1	12	0	0
SE2	1	0	0	2	2	0	**86**	1	2	5	1	0	0	0
S1	4	5	0	4	0	0	1	**83**	0	0	1	0	2	0
S2	0	0	0	0	0	0	0	0	**100**	0	0	0	0	0
S3	0	0	1	0	0	0	8	0	0	**85**	0	1	0	5
L1	0	0	0	9	0	6	0	0	2	0	**70**	13	0	0
H1	2	0	0	0	0	11	0	0	1	0	1	**85**	0	0
H2	0	6	0	0	0	0	0	0	0	0	0	0	**94**	0
N1	0	0	0	0	2	0	0	0	0	0	0	0	0	**98**

V. CONCLUSIONES

En este trabajo se estudian las diferentes técnicas de análisis forense de imágenes para solucionar el problema de la identificación de la fuente de una imagen. Se describe la idea principal de cada una de las técnicas así como algunos de los trabajos más representativos que se han realizado aplicándolas. De acuerdo a la estructura y funcionamiento de las cámaras digitales de dispositivos móviles las técnicas más adecuadas para realizar análisis forense en ellas son las que se basan en el ruido del sensor y las que utilizan las transformadas wavelet. En virtud de lo anterior se propuso un algoritmo para la identificación de los dispositivos móviles fuente combinando las técnicas basadas en la huella del sensor y en la transformación wavelet. Por último con los experimentos realizados y sus resultados se demuestra que la combinación de estas técnicas es efectiva para la identificación del modelo y fabricante con un alto porcentaje de acierto.

Aún estimando que son buenos los resultados obtenidos por la técnica, obviamente existe margen de mejora de las tasas de acierto, sobre todo teniendo en cuenta el caso en el que el número de cámaras aumenta considerablemente. Cuanto mayor sea la mejora en la tasa de acierto mayor será la posibilidad de aplicación de la técnica a situaciones reales. A grandes rasgos las principales líneas de investigación a tener en cuenta en los trabajos futuros son: mejora en la selección del recorte de la fotografía (distintas dimensiones y zonas), optimización de los parámetros de configuración de la máquina SVM, optimización en la selección de la función wavelet y la combinación de esta técnica con otras como las basadas en las caracterísitcas del color, las basadas en las métricas de calidad de la imagen o las que utilizan otros tipos de características extraídas del ruido del sensor.

REFERENCIAS

[1] T. Gloe, M. Kirchner, A. Winkler, and R. Bohme, "Can We Trust Digital Image Forensics?" in *Proceedings of the 15th International Conference on Multimedia*, September 2007, pp. 78–86.

[2] T. Ahonen and A. Moore, "Tomi Ahonen Almanac 2012: Mobile Telecoms Industry Annual Review," 2012. [Online]. Available: http://www.tomiahonen.com/ebook/almanac.html

[3] M. Al-Zarouni, "Mobile Handset Forensic Evidence: a Challenge for Law Enforcement," in *Proceedings of the 4th Australian Digital Forensics Conference*, December 2006.

[4] T. Van Lanh, K. S. Chong, S. Emmanuel, and M. S. Kankanhalli, "A Survey on Digital Camera Image Forensic Methods," in *Proceedings of the IEEE International Conference on Multimedia and Expo*, July 2007, pp. 16–19.

[5] Z. J. Geradts, J. Bijhold, M. Kieft, K. Kurosawa, K. Kuroki, and N. Saitoh, "Methods for Identification of Images Acquired with Digital Cameras," in *Proceedings of the Enabling Technologies for Law Enforcement and Security Conference*, vol. 4232, February 2001, pp. 505–512.

[6] J. Lukas, J. Fridrich, and M. Goljan, "Digital Camera Identification from Sensor Pattern Noise," *IEEE Transactions on Information Forensics and Security*, vol. 1, no. 2, pp. 205–214, June 2006.

[7] F. D. O. Costa, M. Eckmann, W. J. Scheirer, and A. Rocha, "Open Set Source Camera Attribution," in *Proceedings of the 25th Conference on Graphics, Patterns and Images*, August 2012, pp. 71–78.

[8] C. T. Li, "Source Camera Linking Using eEnhanced Sensor Pattern Noise Extracted from Images," in *Proceedings of the 3rd International Conference on Crime Detection and Prevention (ICDP 2009)*. Curran Associates, Inc., December 2009, pp. 1–6.

[9] M. Chen, J. Fridrich, M. Goljan, and J. Lukas, "Determining Image Origin and Integrity Using Sensor Noise," *IEEE Transactions on Information Forensics and Security*, vol. 3, no. 1, pp. 74–90, March 2008.

[10] O. Celiktutan, B. Sankur, and I. Avcibas, "Blind Identification of Source Cell-Phone Model," *IEEE Transactions on Information Forensics and Security*, vol. 3, no. 3, pp. 553–566, September 2008.

[11] C. McKay, "Forensic Analysis of Digital Imaging Devices," University of Maryland," Technical Report, 2007.

[12] C. T. Li and R. Satta, "On the Location-Dependent Quality of the Sensor Pattern Noise and its Implication in Multimedia Forensics," in *Proceedings of the 4th International Conference on Imaging for Crime Detection and Prevention 2011 (ICDP 2011)*. Curran Associates, Inc., November 2011, pp. 1–6.

Expert Systems with Applications 42 (2015) 1927–1940

Contents lists available at ScienceDirect

Expert Systems with Applications

journal homepage: www.elsevier.com/locate/eswa

Smartphone image clustering

Luis Javier García Villalba*, Ana Lucila Sandoval Orozco, Jocelin Rosales Corripio

Group of Analysis, Security and Systems (GASS), Department of Software Engineering and Artificial Intelligence (DISIA), School of Computer Science, Office 431, Universidad Complutense de Madrid (UCM), Calle Profesor José García Santesmases s/n, Ciudad Universitaria, 28040 Madrid, Spain

ARTICLE INFO

Article history:
Available online 25 October 2014

Keywords:
Image clustering
Image forensics analysis
PRNU
Sensor Pattern Noise

ABSTRACT

Every day the use of images from mobile devices as evidence in legal proceedings is more usual and common. Therefore, forensic analysis of mobile device images takes on special importance. This paper explores the branch of forensic analysis which is based on the identification of the source, specifically on the grouping or clustering of images according to their source acquisition. In contrast with other state of the art techniques for source identification, hierarchical clustering does not involve a priori knowledge of the number of images or devices to be identified or training data for a future classification stage. That is, a grouping by classes with all the input images is performed. The proposal is based on the combination of hierarchical and flat clustering and the use of *Sensor Pattern Noise* (SPN). There has been a series of experiments which emulate similar situations to those that may occur in reality to test the robustness and reliability of the results of the technique. The results are satisfactory in all the experiments, obtaining high rates of success.

1. Introduction

At present, the number of cameras integrated into mobile devices has proliferated, allowing millions of consumers to take photographs and even easily share captured content. The mobile industry has developed the technology to reduce costs and thus make them very accessible to the public.

The accessibility and easy use of mobile cameras has the conse quence that a large number of photos are taken with them gener ating more evidence presented before the law on crimes such as credit card fraud, child pornography, industrial espionage, public safety, street violence, etc. Therefore, forensic analysis of such images is particularly important in criminal investigations. There are two main branches within digital image forensic analysis: image source acquisition identification and malicious tampering detection. This work focuses on the first branch. Also, since mobile device cameras have some characteristics that make them different from the rest, this work focuses on images from this type of devices.

There are two major approaches regarding source acquisition identification: closed scenarios and open scenarios. A closed sce nario is one in which the image source identification is performed on a specific and known beforehand set of cameras. For this approach a set of images from each camera is normally used to train a classifier and later the image source acquisition under investigation is predicted. The most commonly used technique for the digital imaging classification task is *Support Vector Machine* (SVM), although there are other options, such as the use of neural networks. This work focuses on image source acquisition identifi cation in open scenarios, i.e., the forensic analyst does not know a priori the camera set to which images whose source identifica tion will be identified belong. Obviously, in this type of classifica tion in which data from cameras are not known beforehand, the objective is not to identify the make and model of the images, but to be able to group the different images into disjoint sets in which all their images belong to the same device. This approach is very close to real life situations, since in many cases the set of cameras to which a set of images may belong is completely unknown to the analyst. In addition, it is virtually impossible to have a set of images to train a classifier with all mobile device cam eras existing in the world. In this case, being able to group images into sets that belong to the same device is very useful, as this can provide very valuable and in some cases conclusive information to judicial investigators.

In this paper a clustering algorithm based on Caldelli, Amerini, Picchioni, and Innocenti (2010) is proposed. As elements for classi fication we use a set of features obtained from SPN noise. Broadly speaking, the main difference is that our proposal takes into account the evolutionary process of cluster formation when

* Corresponding author. Tel.: +34 91 394 76 38; fax: +34 91 394 75 47.
E-mail addresses: javiergv@fdi.ucm.es (L.J. García Villalba), asandoval@fdi.ucm.es (A.L. Sandoval Orozco), jocerosa@ucm.es (J.R. Corripio).
URL: http://gass.ucm.es/en/people/javier/ (L.J. García Villalba).

http://dx.doi.org/10.1016/j.eswa.2014.10.018
0957-4174/

calculating the coefficient that determines the cohesion between the elements of the same cluster and separation between different clusters that are being generated.

This work is divided into five sections, the first being this intro duction. Section 2 briefly presents previous work related to foren sic techniques for mobile device image source acquisition identification. The proposed technique is presented in Section 3. The experiments and their results are presented in Section 4. Finally, in Section 5 the conclusions drawn from this work are presented.

2. Related works

Most research on image source acquisition identification focuses on traditional digital cameras or *Digital Still Camera* (DSC); most of these techniques are not valid for mobile device images. The main reason is that most of the techniques are based on directly or indirectly use of sensor features or in the lens of the digital camera. Regarding the sensor, it is the component that is responsible for capturing the light and generate a digital signal according to its intensity. There are currently two types of sensor technologies that meet this latter purpose in digital cameras: glsCCD y *Complementary Metal Oxide Semiconductor* (CMOS). Both types of sensors essentially consist of metal oxide (*Metal Oxide Semiconductor* (MOS)) distributed in a matrix and they work in a similar way. However the key difference is in the way in which pix els are scanned and the way in which the reading of the charges is carried out. *Charge Coupled Device* (CCD) sensors need an additional chip to process the sensor output information; this causes the manufacture of devices to be more costly and the sensors to be big ger. In contrast, CMOS sensors have independent active pixels, as they themselves perform the digitalization, offering speed and reducing the size and cost of the systems that make up a digital camera. Another difference between these two types of sensors is that the pixels in a CCD array capture light simultaneously, which promotes a more uniform output. CMOS sensors generally perform the reading as progressive scan (avoiding the *blooming* effect). CCD sensors are far superior to the CMOS in terms of noise and dynamic range; on the other hand, CMOS sensors are more sensitive to light and behave better in low light conditions. Early CMOS sensors were somewhat worse than CCDs, but nowadays this has been practically corrected. The CCD technology has reached its limit and nowadays the CMOS technology is developing and gradually overcoming their shortcomings. Most of DSCs use CCD sensors, in mobile devices is more common to use sensors CMOS. Even day by day, reducing the quality differences between CCD and CMOS sensors, in the great majority of cases DSCs sensors notably exceed in quality to sensors in mobile devices digital cameras, and this is a strong reason to require specific techniques for image source acquisition source. Likewise to the case of sensor, mobile devices digital camera lenses, in general, are lower of quality than DSCs lenses.

For any type of image classification, either in open or closed sce narios, it is necessary to obtain certain features that allow classifi cation techniques to perform their task. According to Van Lanh, Chong, Emmanuel, and Kankanhalli (2007), four groups of tech niques can be established for this purpose: based on lens aberra tion (Choi, 2006; Choi, Lam, & Wong, 2006; Van, Emmanuel, & Kankanhalli, 2007), based on the *Color Filter Array* (CFA) matrix interpolation (Bayram, Sencar, & Memon, 2006, 2008; Long & Huang, 2006), based on the sensor imperfections (Chen, Fridrich, Goljan, & Lukás, 2008; Costa, Eckmann, Scheirer, & Rocha, 2012; Kang, Li, Qu, & Huang, 2012; Lukas, Fridrich, & Goljan, 2006) and based on the use of image features (Hu, Li, & Zhou, 2010; Mckay, Swaminathan, Gou, & Wu, 2008; Meng, Kong, & You, 2008; Liu et al., 2012; Ozparlak & Avcibas, 2011). Within the latter group a subdivision can be made based on color features, quality features, and wavelet domain statistics. In Sandoval Orozco, Arenas González, Rosales Corripio, García Villalbas, and Hernandez Castro (2013) an overview of this research can be seen.

This work uses techniques based on sensor imperfections, par ticularly those based on the SPN. The main components of image noise are the *Fixed Pattern Noise* (FPN) and the *Photo Response Non Uniformity* (PRNU). There are several sources of imperfections and noise introduced at different stages of the creating pipeline of an image in a digital camera. Even if a uniform and fully lighted picture is taken it is possible to see small changes in the intensity between pixels. This is due to the shot noise is random and, in large part, the pattern noise is deterministic and is kept approximately equal if several pictures of the same scene are taken.

The noise pattern of an image refers to any spatial pattern that does not change from one image to another. It is composed for the spatial noise which is independent of the signal (FPN) and for the spatial noise due to the difference in the response of each pixel to the incident signal (PRNU). The noise pattern structure is shown in Fig. 1.

Noise FPN is generated by the dark current and it also depends on exposure and temperature. Since the FPN is an independent additive noise, some cameras automatically removed by subtract ing a dark frame to generated images.

Noise PRNU is the dominant part of the Sensor Pattern Noise of an image and it is a multiplicative noise dependent. Noise PRNU is mainly formed by noise *Pixel Non Uniformity* (PNU) and by the low frequency defects as *zoom* settings and light refraction in the dust particles and lenses. Noise PNU is the light sensitivity difference between pixels of the sensor array. It is generated by the lack of homogeneity of the silicon wafers and by the imperfections during the sensor manufacturing process. Due to the nature and origin, it is very unlikely that even the sensors from the same wafer have PNU correlated patterns. This noise is not affected by ambient tem perature nor by humidity. Noise PNU is usually more common, complex and significant in CMOS sensors, due to the complexity of pixel array circuitry.

Once you have the features to be used for classification of images we will focus on issues relating to the classification by clus tering. The analysis of clusters, or clustering, aims to group a col lection of objects into representative classes called clusters, without a priori information, in such a way that the objects belong ing to each cluster keep a greater similarity to objects from other clusters.

Image grouping can be performed using supervised or unsuper vised learning techniques. In the first case it is essential to know the device information a priori, i.e., it is clearly identified with the classification in closed scenarios which requires a training stage with the features extracted from the images and a second

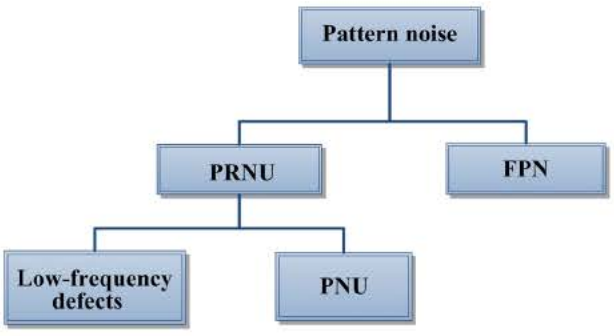


Fig. 1. Sensor Pattern Noise.

classification stage in accordance with the previous result. How ever, in a real case it may be difficult to have the camera in ques tion or a set of photographs taken by it to carry out training, hence the need for unsupervised learning techniques, which directly cor respond to open scenarios.

Traditional clustering has been known to be an unsupervised learning technique; however, there are some cases of supervised clustering where it is possible to apply an anterior or posterior approach to improve the grouping itself. This is to prevent that ele ments of different classes are in the same cluster, which requires having a priori knowledge of the data set. This issue is addressed in Eick, Zeidat, and Zhao (2004), although it is worth mentioning that this article is focused on the use of unsupervised techniques.

In order to determine the similarity between objects belonging to the same cluster, there are distance measures such as Euclidean distance, Manhattan distance, and Chebychev distance, among oth ers. Alternatively, it is possible to use similarity functions $S(X_i, X_j)$ which compare two vectors X_i and X_j symmetrically, i.e., $S(X_i, X_j)$ $S(X_j, X_i)$. These functions reach their highest values as X_i and X_j are more similar. One of the most commonly used mea sures in image source identification is normalized correlation (Bloy, 2008; Caldelli et al., 2010; Fridrich, 2009; Li, 2010b) defined as:

$$corr(X_i, X_j) \quad \frac{(X_i \quad \overline{X_i}) \odot (X_j \quad \overline{X_j})}{\|X_i \quad \overline{X_i}\| \cdot \|X_j \quad \overline{X_j}\|} \tag{1}$$

where $\overline{X_i}$ and $\overline{X_j}$ represent the mean vector, $X_i \odot X_j$ is the scalar product of two vectors and $\|X_i\|$ is the L_2 norm of X_i.

According to the clustering algorithms classification proposed in Rokach (2010), we find the hierarchical methods whose purpose is to achieve a structure called dendrogram which represents the grouping of objects according to their levels of similarity. This grouping can be done in different ways: agglomerative or decisive. Agglomerative grouping initially considers each object as a sepa rate class until iteratively grouping all the objects in a single class. Divisive clustering is based on the idea of starting from a single class until managing to separate all objects into individual classes. There are also partitioning algorithms, wherein starting a partition, the algorithm takes care of moving objects from one cluster to another to minimize certain error criterion. Within this category, the most famous method is k means; however, most of these methods require knowing in advance the number of clusters, which is why they are not widely used in forensic image analysis. Finally, there are other clustering algorithms such as: Zahn (1971) which produces clusters by means of graphs, (Banfield & Raftery, 1993) based on the density where the points within a cluster are given by a certain probability function, clusters based on models such as decision trees (Fisher, 1987) or neural networks (Vesanto & Alhoniemi, 2000) and clustering with soft computing methods such as fuzzy clustering (Hoppner, 1999), evolutionary clustering methods and simulated annealing clustering (Selim & Alsultan, 1991). Xu and Wunsch (2005) shows a comprehensive review of the different types of clustering algorithms, as well as an extensive review of approaches used on this subject in the state of the art. Among other aspects, it is concluded that there is not a universal clustering algorithm to solve any kind of problem and therefore approaches to clustering for each field or situation may be com pletely different. It also highlights the importance of the stage of selection and extraction of the characteristics of the elements to be classified.

There are previous works on image grouping by unsupervised methods; all of them consider SPN as the most reliable criterion for representing a device's digital footprint, hence the SPN is used specifically as a footprint and normalized correlation as a similarity measure to achieve image grouping by device.

Once seeing an overview of the different types of clustering algorithms, then will present some of the related works that deal with the clustering of images using the SPN.

Liu, Lee, Hu, and Choi (2010) uses a classification technique with unsupervised learning where grouping is achieved by graph maximization. Clustering is performed from not oriented graph with weights, starting with an affinity matrix where the connec tion weights between vertices is the correlation value between each SPN, starting with a random node. In each iteration, the remaining nodes are connected and the nodes closest to the central one are chosen, obtaining a new affinity matrix in each step; the algorithm stops when the number of closest nodes is less than a k parameter. Subsequently, the graph is partitioned to the point where similarity in a set is maximum and minimum with respect to other sets.

In Li (2010b) clusters are performed using Markov random fields. A clustering algorithm based on matrix containing all the correlations between the SPN of several cameras is proposed. In each iteration the algorithm groups within classes the most sim ilar SPNs making use of the local features of Markov random fields and assigns a new class label to each SPN maximizing a probability function, the criterion to stop the algorithm is satis fied when there are no label changes after a certain number of iterations.

The algorithm proposed in Caldelli et al. (2010) and on which this research is based uses hierarchical clustering to group images. Prior to the clustering algorithm, the authors apply a function for sensor noise improvement, which strengthens the lower compo nents and attenuates the high components in the wavelet domain in order to remove the scene details in it. With a similarity matrix containing all the correlations between different SPNs and taking as a starting point each image as a single cluster, the clustering algorithm groups the two clusters with the highest correlation value forming a single cluster and updates the matrix with a new row and column that replace the rows and columns of the grouped clusters. The link criterion chosen to mix two clusters was average linkage. In each iteration of the algorithm, cluster status at that time is stored on a partition and the global silhouette coefficient is calcu lated. At the end of the algorithm the partition whose silhouette coefficient value is the lowest is chosen, the number of clusters at that point should correspond to the number of devices that exist initially, as well as the content of each cluster to the SPN for each device. The authors carry out a training stage with the described algorithm and a classification stage for the remaining images, for this it is sufficient to obtain the average of the SPNs for each cluster and compare them against the remaining images, the image will be classified within the cluster whose correlation is highest.

3. Technique description

The proposed unsupervised clustering algorithm is based on the one proposed in Caldelli et al. (2010). It is a combination of a hier archical clustering, and a flat clustering. That is, despite forming a dendrogram structure with each iteration of the algorithm, at the end the clusters are taken as unrelated entities since each of them must correspond to a specific device. The general structure of the proposed clustering algorithm is shown in Fig. 2 (N is de number of images and q is the number of iteration and it begins in 0).

Prior to performing the clustering, it is necessary to obtain SPNs of the image set I using the extraction algorithm and the parameter of noise suppression s_0 5 proposed in Lukas et al. (2006):

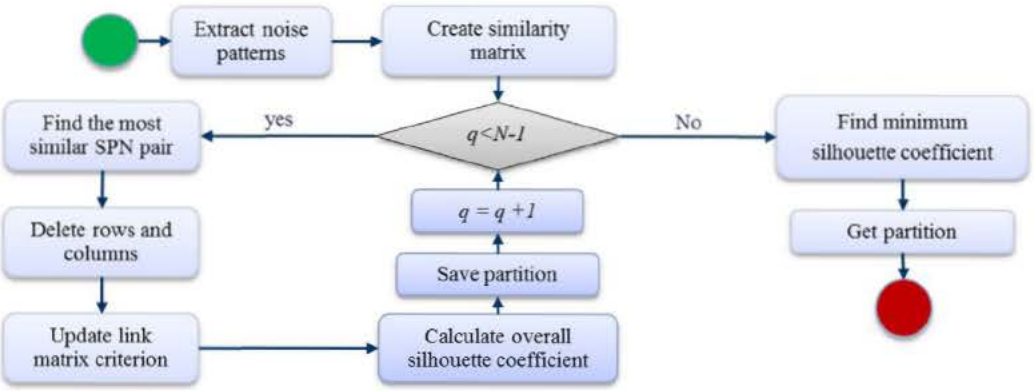


Fig. 2. Clustering algorithm structure.

$$n^{(i)} \quad I^{(i)} \quad F\left(I^{(i)}\right) \tag{2}$$

where $i \quad 1,\ldots,N, N$ is the number of images, $n^{(i)}$ is the noise pat tern of each image i, $I^{(i)}$ is the image with sensor noise of each image i and F is the noise removal filter based on wavelet transform. For this, the algorithm developed in Goljan, Fridrich, and Filler (2009) was used. No noise improvement algorithm, such as those proposed by Caldelli et al. (2010) and Li (2010b), has been used in our pro posal. The Wiener filter in the frequency domain is sufficient to remove most of the scene details that are present when extracting the SPN.

For each of the N noises $(n_1,\ldots,n_N)$ the correlation value is obtained using Eq. (1) and this generates a similarity matrix H of $N \times N$. This matrix is symmetric and consists of ones in its main diagonal (since the correlation of noise with itself is 1). Once the matrix has been generated it will not be necessary to recalculate the correlations between noises along the clustering algorithm, saving time and processing power.

The selected hierarchical clustering algorithm involves finding within the H matrix the noise pair k and l with a highest correlation value. It is worth mentioning that the correlation values in the main diagonal are not taken into account. Then the rows and col umns k and l are deleted and both a new row and a new column are added to the matrix. These new row and column values are the result of a linkage criterion. The function chosen for this work was the average linkage method since its results are more satisfac tory than with other linkage methods such as single linkage or complete linkage, as is suggested in Caldelli et al. (2010). Eq. (3) shows the function of the average linkage method between two clusters A and B.

$$H(A,B) \quad \frac{1}{\|A\|\|B\|}\sum_{n_i\in A, n_j\in B} corr(n_i,n_j) \tag{3}$$

where the $corr(n_i,n_j)$ value is calculated with Eq. (1) and can be taken from the matrix H to simplify the computational processing. $\|A\|$ and $\|B\|$ is the cardinality of the A and B clusters respectively.

Each iteration of the algorithm takes the two clusters with the highest correlation value in the matrix and mixes the objects con tained in them to create a new cluster, while storing the state of the different clusters in partition $P_0,\ldots,P_{N\ 1}$ with the aim of knowing the contents of the cluster at any time. In the hierarchical clustering, the final result of the algorithm is a cluster containing all objects. However, in this work each cluster should represent a device at the end of the execution. For this reason, the silhouette coefficient as a measure of validation of clusters was used. The silhouette coefficient measures the similarity index between the elements of a single cluster (cohesion) and the similarity between the elements of a cluster with respect to the others (separation). Unlike Caldelli et al. (2010), in our proposal the calculation of the silhouette coefficient is performed for each cluster contained in the P_i partition and not for each pattern noise, as noted in Eq. (4).

$$s_j \quad \max(b_j) \quad a_j \tag{4}$$

where a_j (cohesion) is the average correlation between all noise pat terns within the c_j cluster. b_j (separation) is the average correlation of noise patterns contained in the c_j cluster with respect to noise patterns in the remaining clusters. The nearest neighboring cluster is taken, namely the one with the highest correlation.

As can be seen the method of calculating the silhouette coeffi cient varies significantly regarding to the proposal of Caldelli et al. (2010). In our proposal for each algorithm iteration as many silhouettes coefficients as exist formed clusters in that iteration are calculated and not as many silhouette coefficients as total images there are. According to the algorithm progresses and the clusters are increasing their number of images, in our proposal fewer sil houettes coefficients are calculated. Furthermore and the most important thing is that while clusters are formed in each algorithm iteration, each cluster is taken as a single entity for the calculation of silhouette coefficient. Therefore do not take into account each of the independent noises that form each cluster, since what it wants to measure is the cohesion and separation between clusters and not between independent images. Once it has the idea of a cluster as a unitary entity to calculate the silhouette coefficient, it is calcu lated for each cluster taking into account the obtaining of the max imum separation from other clusters and a high cohesion of all the elements of the formed cluster, as it can see in the Eq. (4).

For each iteration q of the algorithm a global measure of all the silhouette coefficients calculated from the K clusters is obtained, this is equivalent to averaging the s_j values in q. Eq. (5) shows this calculation.

$$SC_q \quad \frac{1}{K}\sum_{j\ 1}^{K} s_j \tag{5}$$

Upon completion of the hierarchical clustering, the SC_q with the lowest value is searched for, which indicates that the partition P_q^* clusters are at a greater correlation level. The number of clusters at that moment should correspond to the actual number of devices. The aim of storing the partition at each time of the algorithm is to avoid rerunning the clustering because information of all the clus ters in each iteration q is known. Algorithm 1 shows the proposal's pseudocode.

Algorithm 1. Clustering algorithm

① Calculate $n^{(i)}$ of each image where $i \in 1, \ldots, N$;
② Generate the similarity matrix $H \in R^{N \times N}$;
③ **foreach** $q \in 1, \ldots, N-1$ **do**
④ Find cluster $H(k, l)$ with the highest similarity;
⑤ Remove the pair of rows and columns corresponding to clusters k and l;
⑥ Calculate the values of the new cluster using average link criteria and add the row and its corresponding column;
⑦ Determine the overall silhouette coefficient SC_q;
⑧ Store the partition P_q;
⑨ Find the partition where $min_q(SC_q)$;

As mentioned above, the goal of clustering is to group objects in an unsupervised environment (closed scenario); however, in the chosen methodology it is possible to carry out a training stage and a classification stage to reduce computational complexity and therefore to reduce the execution time of the algorithm. For this it is necessary to divide the image set into two subsets: one of I_e training and one of I_c, classification, the training subset is processed by the algorithm previously described to get to the final K clusters that represent each device. A c_j centroid is then calculated for each cluster by averaging all m noise patterns contained in it. Next the correlation value between each SPN from the I_c subset and each c_j, centroid is obtained, the image is then classified into the cluster where there has been the highest correlation value. The proposed classification can be observed in Algorithm 2.

Algorithm 2. Clustering algorithm with training stage

① Calculate the centroid c_j of each cluster where $j \in 1, \ldots, K$ and $c_j = \frac{1}{m}\sum_1^m n_i$;
② Calculate the pattern noise n_i of the classification subset $I_c \subset I$;
③ **foreach** $n_i \in i_c$ **do**
④ Classify n_i in the cluster with the highest correlation
⑤ $f_j = \arg\max_j corr(n_i, c_j)$;

4. Experiments and results

The experiments were performed with a total set of 1350 photographs from 9 different mobile device camera models. The total set contains 150 photographs from each model. 6 devices are from different manufacturers (Apple iPhone 5, Huawei U8815, LG E400, Samsung GTS5830M, Zopo ZP980 and Nokia 800 Lumia) and the 3 remaining devices were manufactured by Sony (Sony ST25a, Sony ST25i and Sony C2105).

All the images were cropped to 1024 × 1024 pixels because the images have different dimensions and working with these at full size it would be computationally more complex. To reduce the degree of error in the grouping all images have a horizontal orientation; it was necessary a 90° rotation images captured in vertical position. The scenes of the photographs were chosen randomly, both indoors and outdoors, and they were also taken at different times and places in order to simulate a more realistic scenario. In the extraction of the noise pattern from all images, the zero mean of rows and columns was used, 3 RGB color channels were converted to a single matrix in grayscale. Additionally, all experiments were conducted using the Wiener filter in the frequency domain. In Fig. 3 a diagram of the preprocessing performed on the images is shown.

To measure the degree of certainty in the results, the true positive rate TPR was used. The mean TPR for each of the following experiments is calculated, computing for each cluster the number of photos that have been well classified (TPR of each cluster) and averaging the TPRs of all the resulting clusters (if there are fewer clusters than devices the average takes into account the number of devices). To calculate the TPR of each cluster, the device that has the largest number of images with respect to the total of images by device needs to be identified within the cluster, that being the predominant device cluster, then calculate the percentage of photos that have been well classified for that device in the cluster. Actually, in the vast majority of cases it can be seen that a cluster is associated with one or more devices, as it can observed in matrices such as the ones in Tables 1–3.

If there are multiple clusters with the same number of photos from a device or a cluster with the same number of photos from several devices and in turn these being the highest, the cluster that is taken as predominant for the device is one chosen among the different options. It may be the case that if there is an extra cluster, a cluster may not be predominant for any device (see Table 2) and its TPR for this cluster is 0. Or there might be one less cluster (see Table 3). In this case the cluster where there are pictures of several devices, it will be associated with the device with more images have in this cluster, so this cluster will be the predominant for this device. In the case that the maximum number of images of various devices in one cluster are equal, it will take any of them. The TPR calculation of each cluster will take into account in each cluster the number of photos of their predominant device. To calculate the final TPR in this case we must make the sum of all TPRs of each cluster and dividing by the total number of devices initially used for classifying (in the case of Table 3 it will be divide by 5). In Tables 1–3 there are examples that illustrate the calculation of the TPR for the three cases that may occur.

In the results of the experiments 3 possible cases are considered: (a) The number of identified clusters is equal to the number of devices, (b) the number of identified clusters is higher than the number of devices, and (c) the number of identified clusters is lower than the number of devices. Although the first case is ideal, in the second case classifications that do not mix different types of devices in a same cluster can be obtained.

We can divide the experiments according to different criteria:

- Comparison between taking the 1024 × 1024 region from the corner or from the center of the photograph.
- Symmetrical or asymmetrical distribution of photographs (same or different number of pictures per device).
- Comparison of grouping among devices from the same manufacturer but different model.
- Train and then carry out a classification.

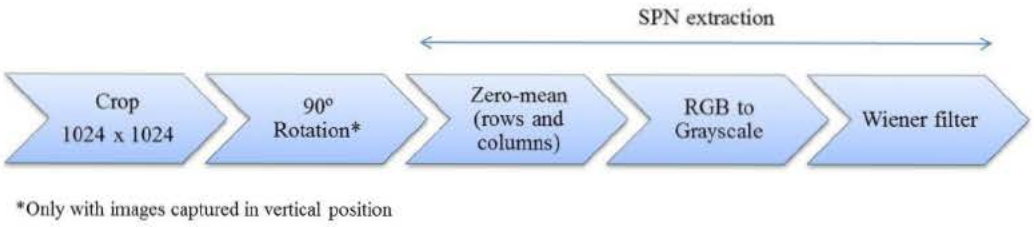


Fig. 3. Image preprocessing scheme.

Table 1
TPR with equal number of devices than cluster.

Brand–model	Clusters					Average
	1	2	3	4	5	TPR
Apple iPhone 5	49	0	0	1	0	
Huawei U8815	0	50	0	0	0	
LG E400	0	1	49	0	0	
Nokia 800 Lumia	0	0	0	50	0	
Samsung GT5830M	0	0	0	0	50	
TPR by cluster	98%	100%	98%	100%	100%	99.2%

Table 2
TPR with less number of devices than clusters.

Brand–model	Clusters				Average
	1	2	3	4	
Apple iPhone 5	100	0	0	0	
Huawei U8815	0	100	0	0	
LG E400	0	0	97	3	
TPR by cluster	100%	100%	97%	0%	99%

Table 3
TPR with more number of devices than clusters.

Brand–model	Clusters				Average
	1	2	3	4	TPR
Apple iPhone 5	100	0	0	0	
Huawei U8815	0	100	0	0	
LG E400	0	0	100	0	
Nokia 800 Lumia	100	0	0	0	
Samsung GT5830M	0	0	0	100	
TPR by cluster	100%	100%	100%	100%	80%

4.1. Comparison between crop corner and crop center

Several experiments were conducted to compare the results between cropping the image from the center or from the upper left corner, this last criterion having a TPR higher (except in the case of 7 devices with 100 images whose difference is minimal). Table 4 shows the TPR according to the different number of devices used and the number of photos used by device. All devices have the same number of photos. Table 4 shows the TPR according to the different number of devices used and the number of photos used by device. All devices have the same number of photos. Table 4 shows that TPR increases in the case of the crop in the center as more devices are grouped, whereas the crop from the corner maintains the good results for the case of 50 images per device and the results are different in the case of 100 images per device. Although Li (2010a) mentions that the areas in the corners are more likely to be saturated and therefore the noise pattern may be affected, the proposed algorithm shows the opposite in the grouping of images.

4.2. Symmetrical clustering

Following is additional information about three of the symmetrical clustering experiments previously conducted whose results generate equal, lower and higher number of clusters than there are devices respectively for their classification. For each experiment, once the algorithm of clustering proposed in this work has been applied, graph will be shown for each generated cluster. This chart shows the degree of correlation of the noise pattern of all the images used in each experiment with respect to the centroid (average of all noise patterns contained in the cluster) of each cluster generated by the algorithm.

The first experiment conducts a symmetrical clustering of 3 devices, 50 images and crop corner. As shown in Table 5 its TPR is 99.33%. Table 5 shows the confusion matrix detailing generated clusters and the images included in each of them.

As can be observed, an equal number of clusters to devices is generated, which in principle is indicative of possibly obtaining a good result. The classification in this case is nearly perfect, with the exception that in cluster 2 there is an image from the E400 LG which should have been classified into cluster 3.

For this experiment Figs. 4 6 show the 3 correlation graphs described above for each of the generated clusters.

In Fig. 4 we can see that the correlations of a device with respect to the remaining correlations are distant, therefore a cluster with images from a single device is properly generated.

In Fig. 5 we can observe that there is an image from the E400 LG whose degree of correlation is not in line with the rest from the same device and those from the Huawei U8815. This specific image is the one which was classified incorrectly, obtaining a lower correlation degree with respect to the centroid than the remaining images of its own cluster, but different from zero and notably higher (on the order of 20 to 400 times higher correlation) to the correlation of the remaining images.

Fig. 6 shows that the pictures of the LG E400 have a similar correlation except for one which practically has a value of 0 correlation with respect to the centroid of cluster 3 and which corresponds to the erroneously classified image.

The second experiment conducts a symmetrical clustering of 5 devices, 50 images and crop center. As can be seen in Table 6 the TPR is 80%. Table 6 shows a confusion matrix detailing generated clusters and the images included in each of them.

As can be observed, a lower number of clusters than of devices is generated, which implies that at least one of the clusters is not pure, i.e. it contains pictures of at least two devices. The classification in this case is completely correct for three of the four generated clusters. In contrast, all the pictures from the Apple iPhone 5 and Nokia 800 Lumia devices are in cluster 1. For this experiment Figs. 7 10 shows the 4 correlation graphs described above for each of the generated clusters.

As shown in Figs. 8 10, which correspond to clusters with all images from a single device, the correlation of the images from the correctly classified device with respect to the other is distant. For these cases the correlation with respect to the centroid of the image outside the cluster is approximately zero in all cases.

Fig. 7 shows that the correlation of images from the Apple iPhone 5 and the Huawei U8815 is similar forming cluster 1 and there is a big difference with the correlation of the rest of the images, which is close to zero.

Table 4
Symmetric clustering TPR in function of the different device number and the number of photos per device.

Number of photos	Crop corner			Crop center		
	3 (%)	5 (%)	7 (%)	3 (%)	5 (%)	7 (%)
50	99.33	99.20	99.71	66.67	80	99.71
100	74.25	100	87.13	66.67	80	87.25

Table 5
Confusion matrix of clustering experiment.

Smartphone	Clusters			Average
	1	2	3	TPR
Apple iPhone 5	50	0	0	
Huawei U8815	0	50	0	
LG E400	0	1	49	
TPR by cluster	100%	100%	98%	99.33%

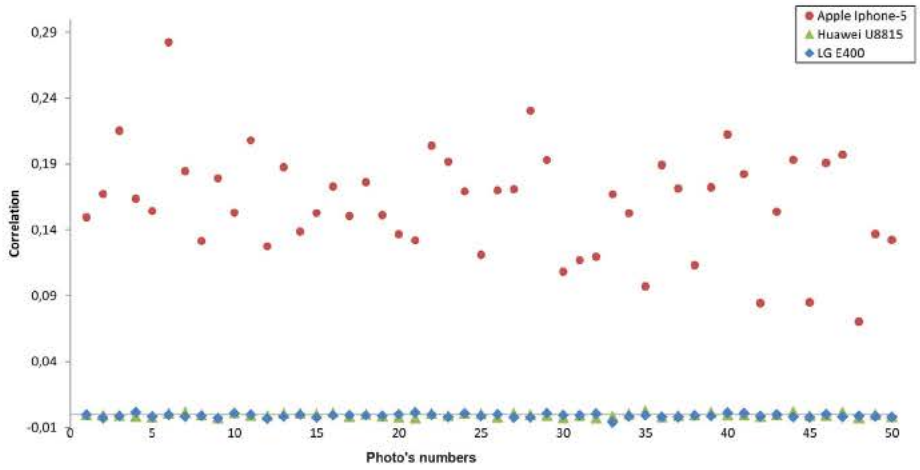


Fig. 4. Cluster 1 correlation graphic respect to the clusters centroid.

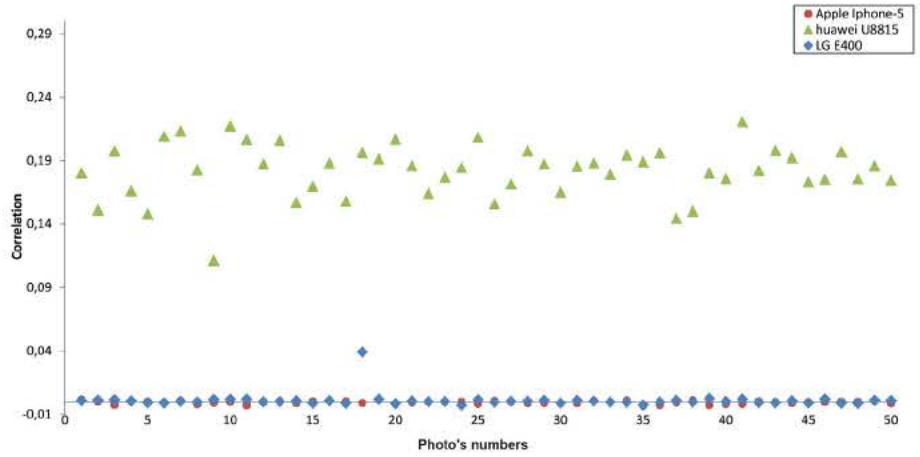


Fig. 5. Cluster 2 correlation graphic respect to the clusters centroid.

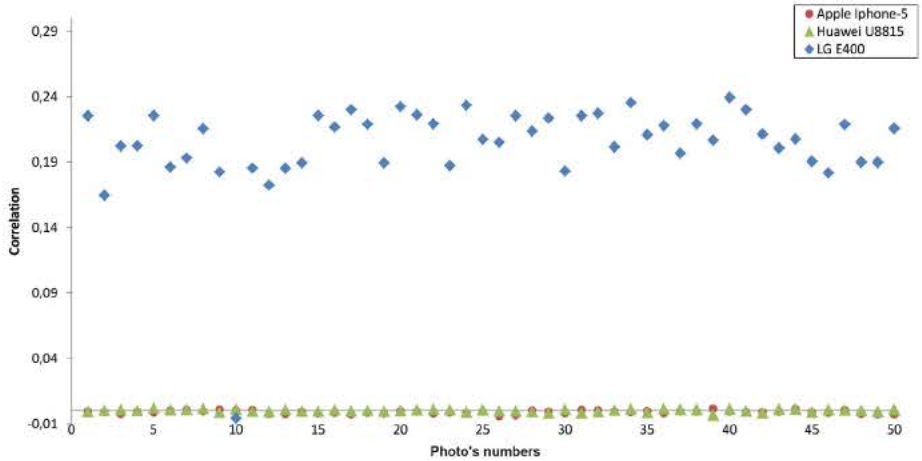


Fig. 6. Cluster 3 correlation graphic respect to the clusters centroid.

The third experiment conducts a symmetrical clustering of 7 devices, 100 images, and crop corner. As can be seen in Table 4, its TPR is 87.13%. Table 7 shows a confusion matrix detailing gen erated clusters and the images included in each of them.

As can be observed, a higher number of clusters than of devices is generated. The classification in this case is completely correct for six of the eight generated clusters. However, clusters 3 and 8 con tain images from the LG E400. While the LG E400 images were divided into two clusters, these only have images from a single device, which is a positive aspect to take into account.

For this experiment Figs. 11 18 shows the 8 correlation graphs described above for each of the generated clusters.

Table 6
Confusion matrix of clustering experiment.

Smartphone	Clusters				Average
	1	2	3	4	TPR
Apple iPhone 5	50	0	0	0	
Huawei U8815	0	50	0	0	
LG E400	0	0	50	0	
Nokia 800 Lumia	50	0	0	0	
Samsung GT5830M	0	0	0	50	
TPR by cluster	100%	100%	100%	100%	80%

As it can be seen in Figs. 11 16, which correspond to clusters that were generated correctly, the correlation of the classified device with respect to the others is distant. For these cases the correlation with respect to the centroid of the image outside the cluster approaches zero, as in previous experiments.

In Fig. 17 there are 97 images from the LG E400 with a correlation that is significantly higher than those of the rest of the images. There are 3 images of the LG E440 whose correlation is practically 0 and very distant from the rest of the photographs from the same device in this cluster, cluster 8 being formed by these.

In Fig. 18, the result of the grouping of these three images on a separate cluster can be observed. The correlation of these three images is significantly higher than that of the rest of the images, which have near zero correlations.

4.3. Asymmetrical clustering

In a closed scenario, it is not very likely to have the same number of images from each device to identify, for that reason experiments were conducted where the sets of images for each device do not possess the algorithm proposed in a real scenario. Sets of images for each device do not possess the algorithm proposed in a real scenario. The results of grouping images from 5 and 7 devices respectively are presented in Tables 8 and 9. The number of images per device is varied and we can still observe a high degree of success (97.28% average TPR of the experiments in Tables 8 and 9).

As can be seen, in the cases of an asymmetrical number of images there have been experiments with groups with significant numerical disparity and in some cases with small groups (5 images from a type of device), yet there have been successful grouping results.

4.4. Same manufacturer different models clustering

Also, experiments were conducted to test the proposal in a scenario where all devices are from the same manufacturer but differ

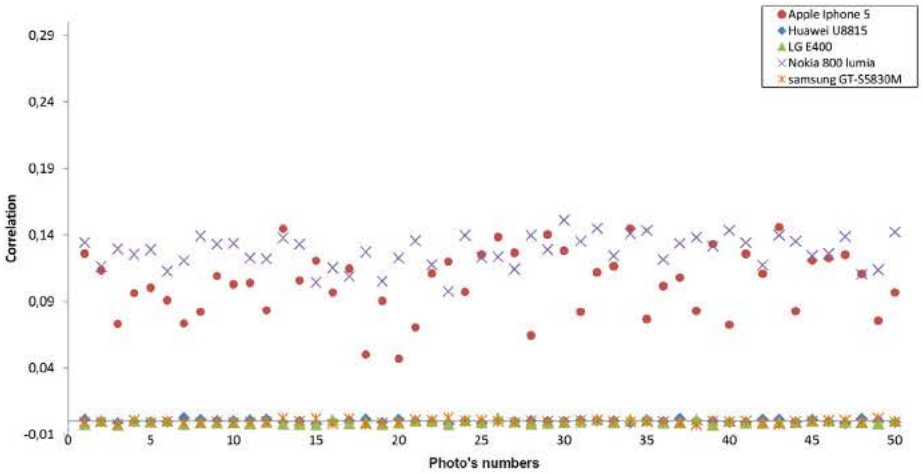


Fig. 7. Cluster 1 correlation graphic respect to the clusters centroid.

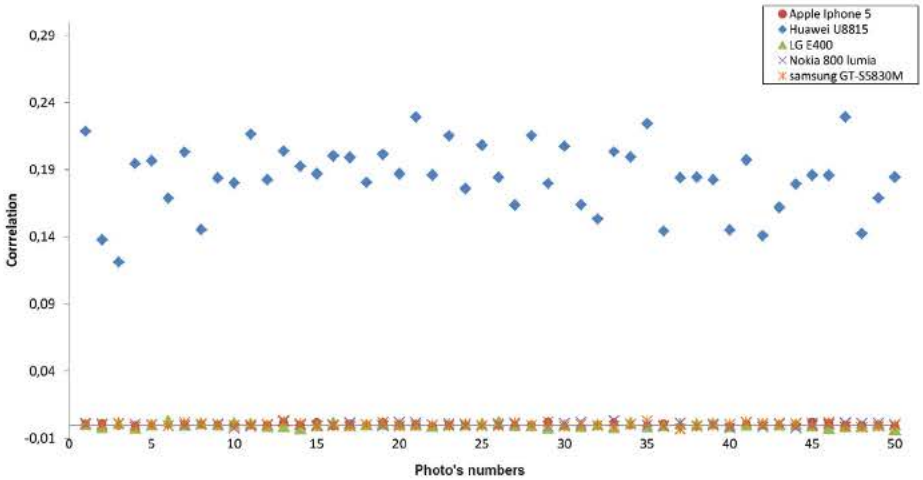


Fig. 8. Cluster 2 correlation graphic respect to the clusters centroid.

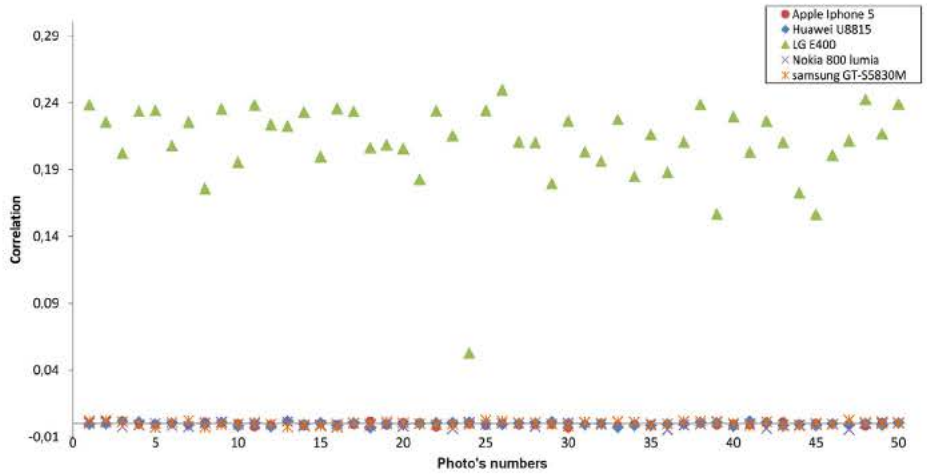


Fig. 9. Cluster 3 correlation graphic respect to the clusters centroid.

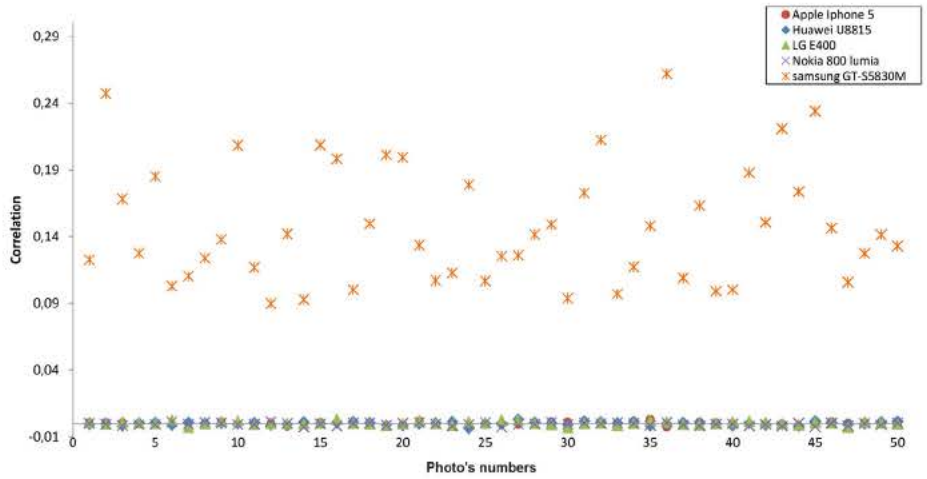


Fig. 10. Cluster 4 correlation graphic respect to the clusters centroid.

Table 7
Confusion matrix of clustering experiment.

Smartphone	Clusters								Average
	1	2	3	4	5	6	7	8	TPR
Apple iPhone 5	100	0	0	0	0	0	0	0	
Huawei U8815	0	100	0	0	0	0	0	0	
LG E400	0	0	97	0	0	0	0	3	
Nokia 800 Lumia	0	0	0	100	0	0	0	0	
Samsung GT5830M	0	0	0	0	100	0	0	0	
Sony ST25A	0	0	0	0	0	100	0	0	
Zopo ZP980	0	0	0	0	0	0	100	0	
TPR by cluster	100%	100%	97%	100%	100%	100%	100%	0%	87.13%

ent models. Camera phones from the same manufacturer should be very similar for many of their products and therefore the sensor noise extracted from different models should be similar. However, Table 10 shows the classification TPR, concluding that for this experiment the correlation value between several SPNs varies enough among different models to identify each one separately, even when the models are very similar as is the case of the ST25a and ST25i models.

4.5. Clustering with training stage

Although clustering methods do not possess information about the data sets to group, some classification experiments were conducted on a set of 5 different devices with different sets of training both with symmetrical and asymmetrical distributions. In what refers strictly to the training stage the proposal (Caldelli et al., 2010) is used. This way computational complexity is widely

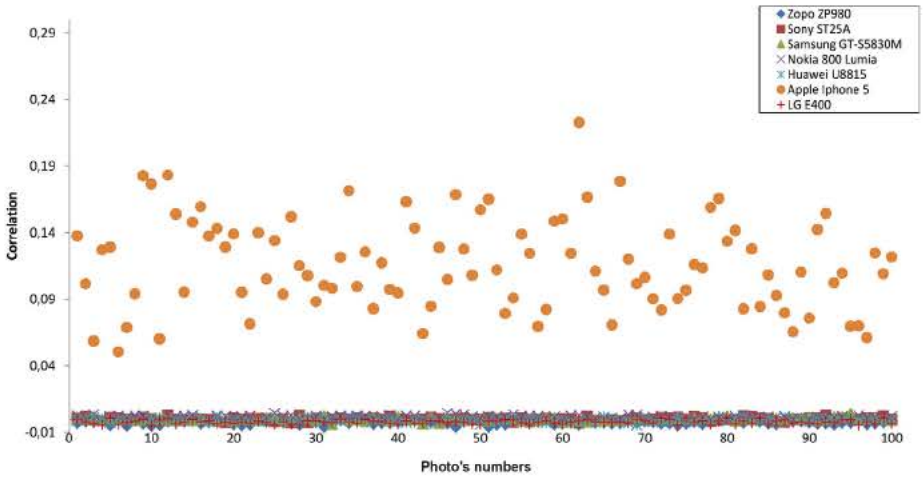


Fig. 11. Cluster 1 correlation graphic respect to the clusters centroid.

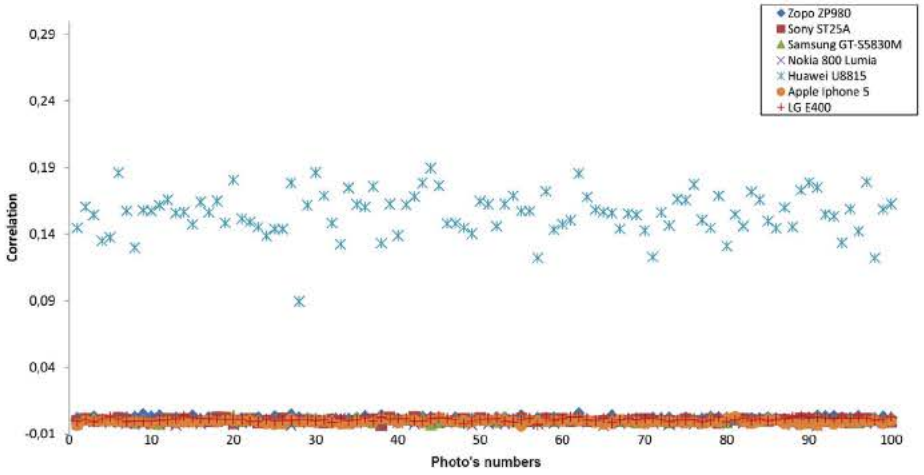


Fig. 12. Cluster 2 correlation graphic respect to the clusters centroid.

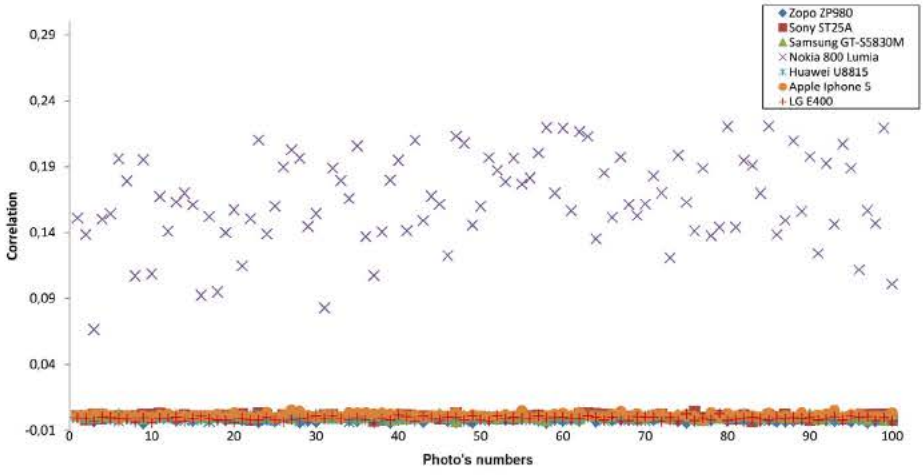


Fig. 13. Cluster 4 correlation graphic respect to the clusters centroid.

reduced at the time of calculating the similarity matrix, this being what takes more time in the execution of the algorithm. Table 11, shows the number of images used for the training stage. In all experiments the remaining images were classified to 150, which the total set of images from each device possesses, no image in the training set is in the classification set and vice versa. Good results are maintained even with asymmetric image sets (98.3% of TPR for the 4 experiments).

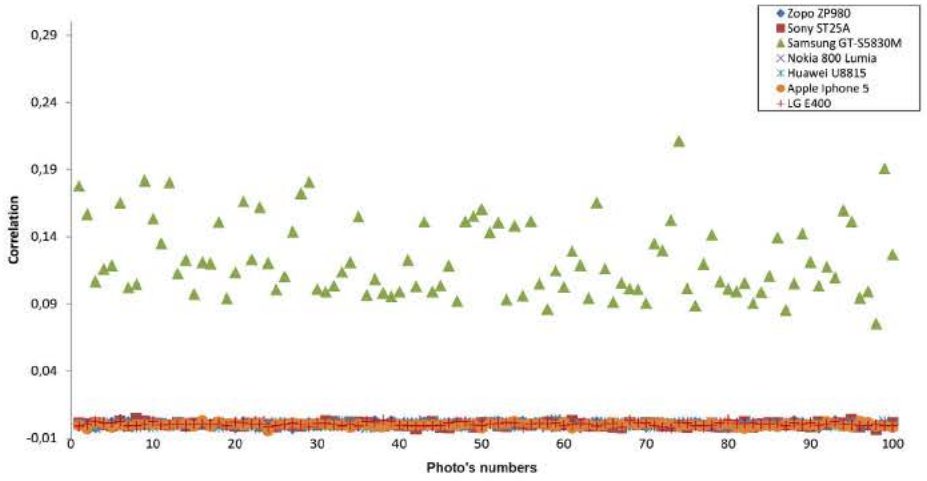


Fig. 14. Cluster 5 correlation graphic respect to the clusters centroid.

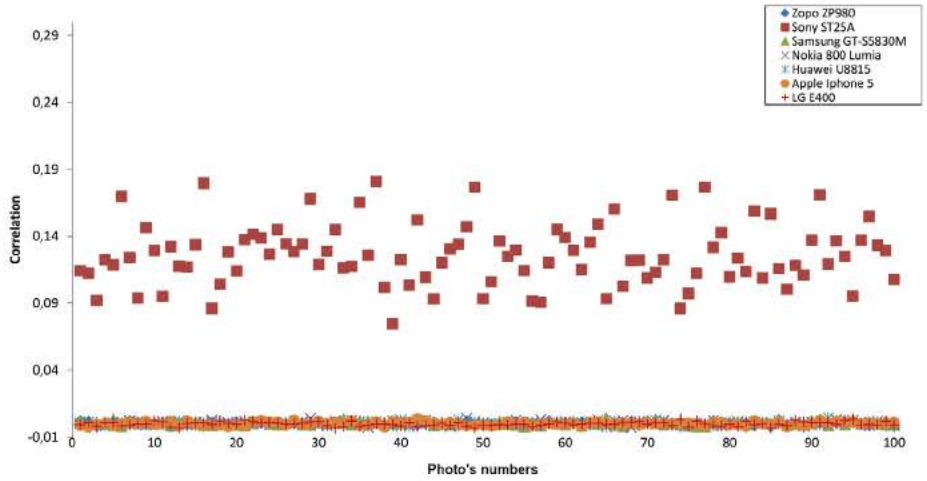


Fig. 15. Cluster 6 correlation graphic respect to the clusters centroid.

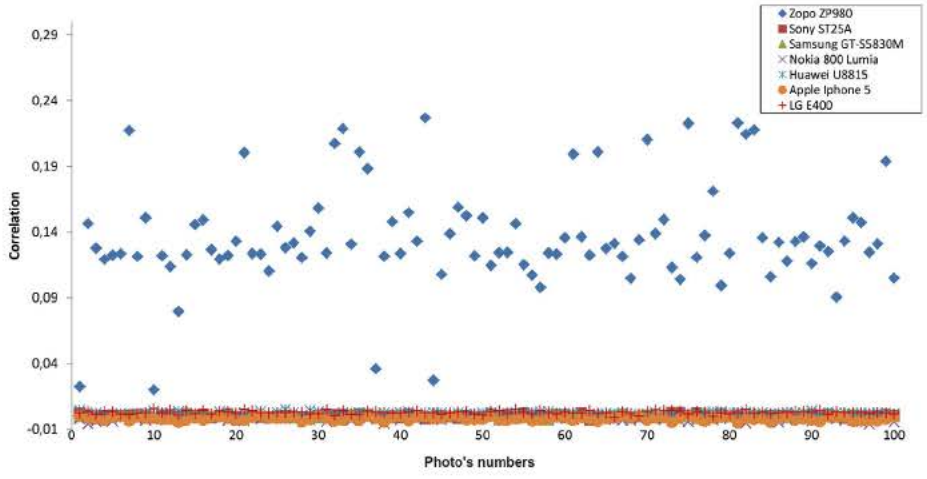


Fig. 16. Cluster 7 correlation graphic respect to the clusters centroid.

4.6. Clustering algorithm execution times

Table 12 shows the clustering algorithm execution times, regardless of the features extraction, of some of the concrete experiments. Only in cases where it has made the training stage, Table 12 shows the total number of images used for each device detailing how many images are used for training stage and how many images are used for classification. These experiments' sample

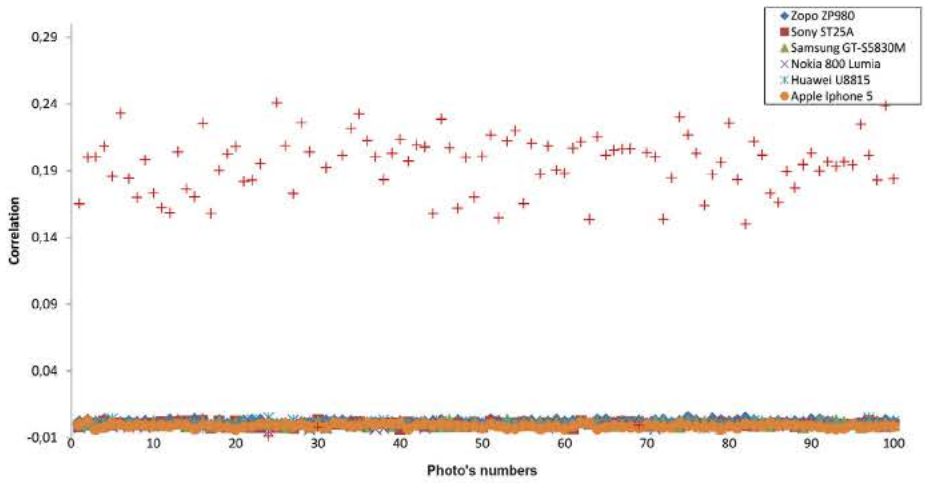


Fig. 17. Cluster 3 correlation graphic respect to the clusters centroid.

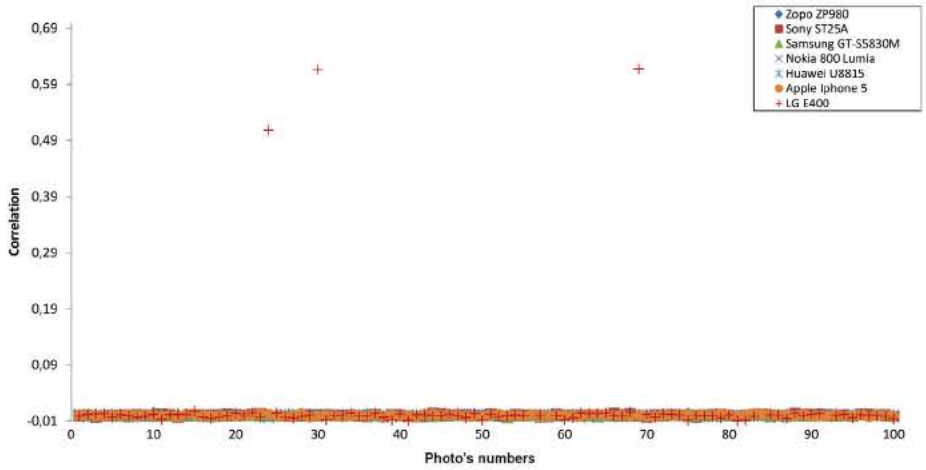


Fig. 18. Cluster 8 correlation graphic respect to the clusters centroid.

Table 8
Asymmetric clustering TPR for 5 devices.

Group	Apple iPhone 5	Huawei U8815	LG E400	Nokia 800 Lumia	Samsung GT5830M	TPR (%)
A	50	50	50	50	50	99.20
B	100	100	100	100	100	100
C	100	95	90	85	80	99.78
D	50	45	40	35	30	99.1
E	100	75	50	25	10	99.6
F	100	30	20	10	5	99

Table 9
Asymmetric clustering TPR for 7 devices.

Group	Apple iPhone 5	Huawei U8815	LG E400	Nokia 800 Lumia	Samsung GT5830M	Sony ST25a	Zopo Zp980	TPR (%)
A	50	50	50	50	50	50	50	99.71
B	100	100	100	100	100	100	100	87.13
C	100	95	90	85	80	75	70	99.84
D	50	45	40	35	30	25	20	99.36
E	100	75	50	25	10	5	1	85.43
F	100	50	40	30	20	10	5	99.21

Table 10
Asymmetric clustering TPR for 3 devices of the same brand.

Models	TPR	
Sony Ericsson	50	100
C2105 ST25a ST25i	99.33%	74.50%

Table 11
Clustering for 5 devices with training stage.

Group	Apple iPhone 5	Huawei U8815	LG E400	Nokia 800 Lumia	Samsung GT5830M	TPR (%)
A	50	50	50	50	50	98.67
B	100	100	100	100	100	99.33
C	100	75	50	25	10	96.67
D	50	45	40	35	30	98.53

times can offer a general idea of the order in runtime of the algo rithm. All images were cropped to 1024×1024. The crew where the experiments were conducted is an Intel Core i7 2670QM 2.2 GHz with 6 Gb RAM and Linux operating system.

Table 12
Execution times.

Clustering	Number of devices	Number of photos (train-classification)	Train stage	Time (s)
Symmetric	3	50	No	445
		100	No	1698
	5	50	No	1207
		150(100–50)	Yes	516
		100	No	4789
		150(50–100)	Yes	1222
Asymmetric	5	50, 45, 40, 35, 30	No	742
		100, 95, 90, 85, 80	No	4299
	7	50, 45, 40, 35, 30, 25, 20	No	1484
		100, 95, 90, 85, 80, 75, 70	No	8992

5. Conclusions

This paper has made an analysis of the main unsupervised image grouping techniques, which are of utmost importance in digital image forensic analysis. Despite the rise of mobile device cameras these days, there are still few references for unsupervised mobile device image grouping in the state of the art. Most of the works refer to the supervised classification and in many cases they are not focused on mobile device images, which have unique characteristics. The noise added in every photograph by the camera sensor, due to the faults in its manufacturing process or defects from daily use, has proven to be a reliable source of device identification. Likewise, the calculation of normalized correlation between sensor noises extracted from two or more pictures is also a measure of similarity commonly used in unsupervised image learning techniques, clustering techniques being the ones which obtain the best results.

The algorithm of this proposal is based on the combination of a hierarchical clustering and a flat clustering for the separation between clusters. The use of the silhouette coefficient for cluster validation proved to report good results when obtaining high TPRs; also, the number of clusters corresponded to the number of actual devices in most cases.

The percentage of correct hits when using image cropping from the left corner obtains better results than when the clipping is centered in the image, despite finding different observations in the literature arguing the saturation and lack of lighting found in those regions.

It was also important to have experimented with different device models from the same manufacturer because, along with the high rate of correct hits by using symmetric and asymmetric distributions of images by device, it checks the adaptability of the algorithm to be applied in a real case.

Experiments conducted in this work have revealed a great diversity of situations with regard to the symmetry or not of the photo sets, their size, the number of devices used and the use of devices of the same brand. After all the experiments, it is concluded that the results of the application of the technique are good (92.98% TPR on average for all the experiments).

Finally we will expose the future lines of work of this work. Firstly the first goal is to improve the TPR results of the algorithm. Within this first aim marked as priority two goals: to generate clusters with images of a single device and as far as possible to set the maximum difference between the number generated clusters by the proposed algorithm and the number of initial devices. The second line of work will focus on using other image features different from those of SPN for classifying images, but following using the same clustering algorithm. For this purpose, we will study *wavelets*, color and *Image Quality Metrics* (IQM) features. The third goal is to create a final stage, once the execution of the proposed clustering algorithm finished, for verifying the generated clusters and if it is necessary make the appropriate changes in the cluster to improve the results. The fourth and final aim is to apply these techniques to digital videos generated with mobile devices.

Acknowledgements

Part of the computations of this work were performed in EOLO, the HPC of Climate Change of the International Campus of Excellence of Moncloa, funded by MECD and MICINN. This is a contribution to CEI Moncloa.

References

Banfield, J. D., & Raftery, A. E. (1993). Model-based gaussian and non-gaussian clustering. *Biometrics, 49*(3), pp–821.

Bayram, S., Sencar, H. T., & Memon, N. (2006). Improvements on source camera-model identification based on CFA interpolation. In *Proceedings of the international conference on digital forensics* (pp. 24–27). Springer.

Bayram, S., Sencar, H. T., & Memon, N. (2008). Classification of digital camera-models based on demosaicing artifacts. *Digital Investigation, 5*(1–2), 49–59.

Bloy, G. J. (2008). Blind camera fingerprinting and image clustering. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 30*(3), 532–535.

Caldelli, R., Amerini, I., Picchioni, F., & Innocenti, M. (2010). Fast image clustering of unknown source images. In *Proceedings of the IEEE international workshop on information forensics and security* (pp. 1–5). IEEE.

Chen, M., Fridrich, J., Goljan, M., & Lukás, J. (2008). Determining image origin and integrity using sensor noise. *IEEE Transactions on Information Forensics and Security, 3*(1), 74–90.

Choi, K.S. (2006). Source Camera Identification using footprints from lens aberration. In *Proceedings on digital photography II. Proceedings of the SPIE international society for optical engineering; number 852 in 6069* (pp. 60690J–60690J–8).

Choi, K. S., Lam, E. Y., & Wong, K. Y. (2006). Automatic source camera identification using the intrinsic lens radial distortion. *Optics Express, 14*(24), 11551–11565.

Costa, F. D. O., Eckmann, M., Scheirer, W. J., & Rocha, A. (2012). Open set source camera attribution. In *Proceedings of the 25th conference on graphics, patterns and images* (pp. 71–78). IEEE.

Eick, C. F., Zeidat, N., & Zhao, Z. (2004). Supervised clustering-algorithms and benefits. In *Proceedings of the IEEE international conference on tools with artificial intelligence* (pp. 774–776). IEEE.

Fisher, D. H. (1987). Knowledge acquisition via incremental conceptual clustering. *Machine Learning, 2*(2), 139–172.

Fridrich, J. (2009). Digital image forensics. *IEEE Signal Processing Magazine, 26*(2), 26–37.

Goljan, M., Fridrich, J., & Filler, T. (2009). Large scale test of sensor fingerprint camera identification. In *Proceedings on media forensics and security. international society for optics and photonics* (Vol. 7254), (pp. 72540I–72540I).

Hoppner, F. (1999). *Fuzzy cluster analysis: Methods for classification, data analysis and image recognition. Jossey-Bass higher and adult education series*. Wiley.

Hu, Y., Li, C.T., & Zhou, C. (2010). Selecting forensic features for robust source camera identification. In *Computer symposium (ICS), 2010 international* (pp. 506–511).

Kang, X., Li, Y., Qu, Z., & Huang, J. (2012). Enhancing source camera identification performance with a camera reference phase sensor pattern noise. *IEEE Transactions on Information Forensics and Security, 7*(2), 393–402.

Li, C. T. (2010a). Source camera identification using enhanced sensor pattern noise. *IEEE Transactions on Information Forensics and Security, 5*(2), 280–287.

Li, C. T. (2010b). Unsupervised classification of digital images using enhanced sensor pattern noise. In *Proceedings of the IEEE international symposium on circuits and systems* (pp. 3429–3432). IEEE.

Liu, Bb., Lee, H. K., Hu, Y., & Choi, C. H. (2010). On classification of source cameras: A graph based approach. In *Proceedings of the IEEE international workshop on information forensics and security* (pp. 1–5). IEEE.

Liu, Q., Li, X., Chen, L., Cho, H., Cooper, A. P., Chen, Z., et al. (2012). Identification of smartphone-image source and manipulation. In H. Jiang, W. Ding, M. Ali, & X. Wu (Eds.), *Advanced research in applied artificial intelligence. Lecture notes in computer science* (Vol. 7345, pp. 262–271). Dalian, China, Berlin, Heidelberg: Springer.

Long, Y., & Huang, Y. (2006). Image based source camera identification using demosaicing. In *Proceedings of the IEEE 8th workshop on multimedia signal processing* (pp. 419–424). IEEE.

Lukas, J., Fridrich, J., & Goljan, M. (2006). Digital camera identification from sensor pattern noise. *IEEE Transactions on Information Forensics and Security, 1*(2), 205–214.

Mckay, C., Swaminathan, A., Gou, H., & Wu, M. (2008). Image Acquisition forensics: Forensic analysis to identify imaging source. In *International conference on acoustics speech and signal processing (ICASSP)* (pp. 1657–1660). IEEE.

Meng, F.J., Kong, X.W., & You, X.G. (2008). Source camera identification based on image bi-coherence and wavelet features. In *Proceedings of the fourth annual IFIP WG 11.9 international conference on digital forensics* (pp. 702–705). Kyoto, Japan.

Ozparlak, L., & Avcibas, I. (2011). Differentiating between images using wavelet-based transforms: A comparative study. *IEEE Transactions on Information Forensics and Security, 6*(4), 1418–1431.

Rokach, L. (2010). A survey of clustering algorithms. In O. Maimon & L. Rokach (Eds.), *Data mining and knowledge discovery handbook* (pp. 269–298). US: Springer.

Sandoval Orozco, A.L., Arenas González, D.M., Rosales Corripio, J., García Villalbas, L.J., & Hernandez-Castro, J.C. (2013). Techniques for source camera identification. In *Proceedings of the 6th international conference on information technology* (pp. 1–9).

Selim, S. Z., & Alsultan, K. (1991). A simulated annealing algorithm for the clustering problem. *Pattern Recognition, 24*(10), 1003–1008.

Van, L.T., Emmanuel, S., & Kankanhalli, M.S. (2007). Identifying source cell phone using chromatic aberration. In *Proceedings of the IEEE international Conference on Multimedia and Expo* (pp. 883–886).

Van Lanh, T., Chong, K. S., Emmanuel, S., & Kankanhalli, M. S. (2007). A survey on digital camera image forensic methods. In *Proceedings of the IEEE international conference on multimedia and expo* (pp. 16–19). IEEE.

Vesanto, J., & Alhoniemi, E. (2000). Clustering of the self-organizing map. *IEEE Transactions on Neural Networks, 11*(3), 586–600.

Xu, R., Wunsch, D., et al. (2005). Survey of clustering algorithms. *IEEE Transactions on Neural Networks, 16*(3), 645–678.

Zahn, C. T. (1971). Graph–theoretical methods for detecting and describing gestalt clusters. *IEEE Transactions on Computers, C-20*(1), 68–86.

The 7th International Conference on Information Technology

Big Data

ISSN 2306-6105

ISBN 978-9957-8583-3-9

Conference Proceeding

Full

Prepared and Edited by:

- ICIT15 General Chair
  - Ali Al-Dahoud, Dean of Science and IT Faculty
- Editorial Board
  - Hani Mimi, ICIT15 Co-chair
  - Khalid Jaber, ICIT5 Co-chair
  - Israa Sabatin, Designer
  - Hanade Al-Shawabkeh, Editor
  - Ayman Al-Qafa'an, Editor

ICIT 2015 The 7th International Conference on Information Technology
doi:10.15849/icit.2015.0014

Unsupervised Classification of Mobile Device Images

Jocelin Rosales Corripio, Ana Lucila Sandoval Orozco, Luis Javier García Villalba
Group of Analysis, Security and Systems (GASS)
Department of Software Engineering and Artificial Intelligence (DISIA)
Faculty of Information Technology and Computer Science, Office 431
Universidad Complutense de Madrid (UCM)
Calle Profesor José García Santesmases, 9
Ciudad Universitaria, 28040 Madrid, Spain
Email: jocelinr@ucm.es, {asandoval, javiergv}@fdi.ucm.es

***Abstract*— As mobile devices are seeing widespread usage in the everyday life, the images from mobile devices can be used as evidence in legal purposes. Accordingly, the identification of mobile devices images are of significant interest in digital forensics. In this paper, we propose a method to determine the mobile devices camera source based on the grouping or clustering of images according to their source acquisition. Our clustering technique does not involve a priori knowledge of the number of images or devices to be identified or training data for a future classification stage. The proposal combines of hierarchical and flat clustering and the use of sensor pattern noise. Experimental results show that our approach is very promising for identifying mobile devices source.**

Keywords*— *Image Clustering; Image Forensics Analysis; PRNU; Sensor Pattern Noise

I. INTRODUCTION

Nowadays, even suffering the impact of global financial crisis, the sales of mobile devices such as cell phones, smartphones or tablets, is still increasing. About 78.1% of mobile phones sold in 2010 have an integrated camera [1]. Integrated cameras in mobile devices outnumber traditional Digital Still Camera (DSCs). The sales of cameras integrated into mobile devices in 2013 exceeded 1800 million units. Similarly, there are predictions that the DSCs will disappear in favour of integrated mobile devices [2], since the quality of these cameras is growing at an unstoppable rate. Also, the emergence of cameras in mobile devices should not only be measured in sales figures, as in our daily life it is common to see how people use photographs from these devices for a variety of situations – personal life, news, legal evidence, software applications and so on. Therefore, forensic analysis of such images is particularly important in criminal investigations.

The image source acquisition identification and malicious tampering detection are of significant interest in digital image forensic analysis. This work focuses on the first branch. Also, since mobile device cameras have some characteristics that make them different from the rest, this work focuses on images from this type of devices. The source acquisition identification has closed scenarios and open scenarios approaches regarding. A closed scenario is one in which the image source identification is performed on a specific and known beforehand set of cameras. In closed scenario approach normally use to train and predict process in order to classify like Support Vector Machine (SVM) classifier. Instead, in open scenarios the forensic analyst does not know a priori the camera set to which images whose source identification will be identified belong.

In this paper, we propose a method that utilizes the hierarchical and flat clustering to image source identification in open scenarios. The objective of this approach is to group the different images into disjoint sets in which all their images belong to the same device. This approach is very close to real-life situations, since in many cases the set of cameras to which a set of images may belong is completely unknown to the analyst. In addition, it is virtually impossible to have a set of images to train a classifier with all mobile device cameras existing in the world. In this case, being able to group images into sets that belong to the same device is very useful, as this can provide very valuable and in some cases conclusive information to judicial investigators. The remainder of this paper is organized as follows. Section 2 briefly presents previous work related to forensic techniques for mobile device image source acquisition identification. The proposed technique is presented in section 3. The experiments and their results are presented in section 4. Finally, in section 5 the conclusions drawn from this work are presented.

A. Image Formation in Digital Cameras

The first step is to understand and create image processing forensic algorithms is to thoroughly know the process of image acquisition in digital cameras. Fig. 1 summarizes this process.

doi:10.15849/icit.2015.0014

Fig. 1. Image acquisition process in digital cameras [3]

First, the lens system captures light from the scene by controlling the exposure, focus, and image stabilization. Next, the light passes through a set of filters that improve the visual quality of the image, and then the light gets to the image sensor called Color Filter Array (CFA); this is an array of light sensitive elements called pixels. Note that the choice of the CFA can influence the sharpness and the final appearance of the image since there are different CFA patterns.

The most commonly used model is the Green-Red-Green-Blue (GRGB) Bayer pattern; other models are: Red-Green-Blue-Emerald (RGBE), Cyan-Yellow-Yellow-Magenta (CYYM), Cyan-Yellow-Green-Magenta (CYGM) or Red-Green-Blue-White (RGBW). The incident light on the colored filters gets to a sensor which is responsible for generating an analogue signal proportional to the intensity of received light, keeping these values in an internal array.

There are currently two types of sensor technologies that meet this latter purpose in digital cameras: CCD (Charge Coupled Device) and CMOS (Complementary Metal Oxide Semiconductor). Both types of sensors essentially consist of Metal Oxide Semiconductors (MOS) and they work in a similar way, although the key difference is in the way in which pixels are scanned and the way in which the reading of the charges is carried out. CCD sensors need an additional chip to process the sensor's output information; this causes the manufacture of devices to be more costly and the sensors to be bigger. In contrast, CMOS sensors have independent active pixels, as they themselves perform the digitalization, offering speed and reducing the size and cost of the systems that make up a digital camera. Another difference between these two types of sensors is that the pixels in a CCD array capture light simultaneously, which promotes a more uniform output. CMOS sensors generally perform the reading as progressive scan (avoiding the blooming effect). CCD sensors are far superior to the CMOS in terms of noise and dynamic range; on the other hand, CMOS sensors are more sensitive to light and behave better in low light conditions. Early CMOS sensors were somewhat worse than CCDs, but nowadays this has been practically corrected.

The CCD technology has reached its limit and it is now when CMOS is being developed and its weaknesses are being overcome, so much that the majority of smartphones contain CMOS sensors. Signals stored by the CCD/CMOS sensor are then converted into a digital signal and transmitted to the image processor, once the image processor receives the digital signal it eliminates noise and other introduced anomalies. Some other processes applied to the signal are color interpolation, gamma correction, and color correction.

II. Previous Works in Image Forensic Analysis

Most research on image source acquisition identification focuses on traditional digital cameras or DSC; most of these techniques are not valid for mobile device images. In [4] an overview of this research can be seen.

For any type of image classification, either in open or closed scenarios, it is necessary to obtain certain features that allow classification techniques to perform their task. According to [3], four groups of techniques can be established for this purpose: based on lens aberration, based on the CFA matrix interpolation, based on the sensor imperfections and based on the use of image features. Within the latter group a subdivision can be made based on color features, quality features, and wavelet domain statistics. This work uses techniques based on sensor imperfections, particularly those based on the sensor pattern noise (SPN). The main components of image noise are the Fixed Pattern Noise (FPN) and the Photo Response Non Uniformity (PRNU). There are several sources of imperfections and noise introduced at different stages of the creating pipeline of an image in a digital camera. Even if a uniform and fully lighted picture is taken it is possible to see small changes in the intensity between pixels. This is due to the shot noise is random and, in large part, the pattern noise is deterministic and is kept approximately equal if several pictures of the same scene are taken.

The analysis of clusters, or clustering, aims to group a collection of objects into representative classes called clusters, without a priori information, in such a way that the objects belonging to each cluster keep a greater similarity to objects from other clusters. Image grouping can be performed using supervised or unsupervised learning techniques. In the first case it is essential to know the device information a priori, i.e., it is clearly identified with the classification in closed scenarios which requires a training stage with the features extracted from the images and a second classification stage in accordance with the previous result. However, in a real case it may be difficult to have the camera in question or a set of photographs taken by it to carry out training, hence the need for unsupervised learning techniques, which directly correspond to open scenarios.

Traditional clustering has been known to be an unsupervised learning technique; however, there are some cases of supervised clustering where it is possible to apply an anterior or posterior approach to improve the grouping itself. This is to prevent that elements of different classes are in the same cluster, which requires having a priori knowledge of the data set. This issue is addressed in [6], although it is worth mentioning that this article is focused on the use of unsupervised techniques.

In order to determine the similarity between objects belonging to the same cluster, there are distance measures such as Euclidean distance, Manhattan distance, and Chebychev distance, among others. Alternatively, it is possible to use

doi:10.15849/icit.2015.0014

similarity functions $S(X_i, X_j)$ which compare two vectors X_i and X_j symmetrically, i.e., $S(X_i, X_j) = S(X_j, X_i)$. These functions reach their highest values as X_i and X_j are more similar. One of the most commonly used measures in image source identification is normalized correlation [7][8][17] defined in equation 1.

$$corr(X_i, X_j) = \frac{(X_i - \overline{X_i}) \odot (X_j - \overline{X_j})}{\|X_i - \overline{X_i}\| \, \|X_j - \overline{X_j}\|} \quad (1)$$

Where $\overline{X_i}$ and $\overline{X_j}$ represent the mean vector, $X_i \odot \overline{X_j}$ is the scalar product of two vectors and $\|X_i\|$ is the L_2 norm of X_i.

According to the clustering algorithms classification proposed in [9], we find the hierarchical methods whose purpose is to achieve a structure called dendogram which represents the grouping of objects according to their levels of similarity. This grouping can be done in different ways: agglomerative or decisive. Agglomerative grouping initially considers each object as a separate class until iteratively grouping all the objects in a single class. Divisive clustering is based on the idea of starting from a single class until managing to separate all objects into individual classes. There are also partitioning algorithms, wherein starting a partition, the algorithm takes care of moving objects from one cluster to another to minimize certain error criterion. Within this category, the most famous method is k-means; however, most of these methods require knowing in advance the number of clusters, which is why they are not widely used in forensic image analysis. Finally, there are other clustering algorithms such as: [10] which produces clusters by means of graphs, [11] based on the density where the points within a cluster are given by a certain probability function, clusters based on models such as decision trees [12] or neural networks [13] and clustering with soft-computing methods such as fuzzy clustering [14], evolutionary clustering methods and simulated annealing clustering [15].

There are previous works on image grouping by unsupervised methods; all of them consider SPN as the most reliable criterion for representing a device's digital footprint, hence the PRNU is used specifically as a footprint and normalized correlation as a similarity measure to achieve image grouping by device.

[16] uses a classification technique with unsupervised learning where grouping is achieved by graph maximization. Clustering is performed from not-oriented graph with weights, starting with an affinity matrix where the connection weights between vertices is the correlation value between each SPN, starting with a random node. In each iteration, the remaining nodes are connected and the nodes closest to the central one are chosen, obtaining a new affinity matrix in each step; the algorithm stops when the number of closest nodes is less than a k parameter. Subsequently, the graph is partitioned to the point where similarity in a set is maximum and minimum with respect to other sets.

In [8] clusters are performed using Markov random fields. A clustering algorithm based on matrix containing all the correlations between the SPN of several cameras is proposed. In each iteration the algorithm groups within classes the most similar SPNs making use of the local features of Markov random fields and assigns a new class label to each SPN maximizing a probability function, the criterion to stop the algorithm is satisfied when there are no label changes after a certain number of iterations.

The algorithm proposed in [17] and on which this research is based uses hierarchical clustering to group images. Prior to the clustering algorithm, the authors apply a function for sensor noise improvement, which strengthens the lower components and attenuates the high components in the wavelet domain in order to remove the scene details in it. With a similarity matrix containing all the correlations between different SPNs and taking as a starting point each image as a single cluster, the clustering algorithm groups the two clusters with the highest correlation value forming a single cluster and updates the matrix with a new row and column that replace the rows and columns of the grouped clusters. The link criterion chosen to mix two clusters was average linkage. In each iteration of the algorithm, cluster status at that time is stored on a partition and the global silhouette coefficient is calculated. At the end of the algorithm the partition whose silhouette coefficient value is the lowest is chosen, the number of clusters at that point should correspond to the number of devices that exist initially, as well as the content of each cluster to the SPN for each device. The authors carry out a training stage with the described algorithm and a classification stage for the remaining images, for this it is sufficient to obtain the average of the SPNs for each cluster and compare them against the remaining images, the image will be classified within the cluster whose correlation is highest.

III. TECHNIQUE DESCRIPTION

The proposed unsupervised clustering algorithm is based on the one proposed in [17]. It is a combination of a hierarchical clustering, and a flat clustering. That is, despite forming a dendrogram structure with each iteration of the algorithm, at the end the clusters are taken as unrelated entities since each of them must correspond to a specific device.

Prior to performing the clustering, it is necessary to obtain sensor pattern noises of the image set I using the extraction algorithm and the parameter of noise suppression $s_0 = 5$ proposed in [5]. Equation 2 shows this calculation.

$$n^{(i)} = I^{(i)} - F(I^{(i)}) \quad (2)$$

Where $i = 1, \dots, N$, N is the number of images, $n^{(i)}$ is the noise pattern of each image i, $I^{(i)}$ is the image with sensor noise of each image i and F is the noise removal filter based on wavelet transform. For this, the algorithm developed by Goljan et al. in [18] was used. No noise improvement algorithm, such as those proposed by [8] and [17], has been used in our proposal. The Wiener filter in the frequency domain is sufficient to remove most of the scene details that are present when extracting the SPN.

For each of the N noises $n_1, \dots, n_N$ the correlation value is obtained using equation 1 and this generates a similarity matrix H of $N \times N$. This matrix is symmetric and consists of ones in

doi:10.15849/icit.2015.0014

its main diagonal (since the correlation of noise with itself is 1). Once the matrix has been generated it will not be necessary to recalculate the correlations between noises along the clustering algorithm, saving time and processing power.

The selected hierarchical clustering algorithm involves finding within the H matrix the noise pair k and l with a highest correlation value. It is worth mentioning that the correlation values in the main diagonal are not taken into account. Then the rows and columns k and l are deleted and both a new row and a new column are added to the matrix. These new row and column values are the result of a linkage criterion. The function chosen for this work was the average linkage method since its results are more satisfactory than with other linkage methods such as single linkage or complete linkage, as is suggested in [17]. Equation 3 shows the function of the average linkage method between two clusters A and B.

$$\mathrm{H(A,B)} = \frac{1}{\|\mathrm{A}\|\|\mathrm{B}\|}\sum_{\mathrm{n_i}\in\mathrm{A},\mathrm{n_j}\in\mathrm{A}} \mathrm{corr}\left(\mathrm{n_i},\mathrm{n_j}\right) \quad (3)$$

where the $corr(n_i, n_j)$ value is calculated with equation 1 and can be taken from the matrix H to simplify the computational processing. $\|A\|$ and $\|B\|$ is the cardinality of the A and B clusters respectively.

Each iteration of the algorithm takes the two clusters with the highest correlation value in the matrix and mixes the objects contained in them to create a new cluster, while storing the state of the different clusters in partition $P_0, \dots, P_{N-1}$ with the aim of knowing the contents of the cluster at any time. In the hierarchical clustering, the final result of the algorithm is a cluster containing all objects. However, in this work each cluster should represent a device at the end of the execution. For this reason, the silhouette coefficient as a measure of validation of clusters was used. The silhouette coefficient measures the similarity index between the elements of a single cluster (cohesion) and the similarity between the elements of a cluster with respect to the others (separation). Unlike Caldelli et al. [17], in our proposal the calculation of the silhouette coefficient is performed for each cluster contained in the P_i partition and not for each pattern noise, as noted in Equation 4.

$$\mathrm{s_j} = \max\left(\mathrm{b_j}\right) - \mathrm{a_j} \quad (4)$$

where a_j (cohesion) is the average correlation between all noise patterns within the c_j cluster. b_j (separation) is the average correlation of noise patterns contained in the c_j cluster with respect to noise patterns in the remaining clusters. The nearest neighboring cluster is taken, namely the one with the highest correlation.

For each iteration q of the algorithm a global measure of all the silhouette coefficients calculated from the K clusters is obtained, this is equivalent to averaging the s_j values in q. Equation 5 shows this calculation.

$$\mathrm{SC_q} = \frac{1}{\mathrm{K}}\sum_{\mathrm{j}=1}^{\mathrm{K}} \mathrm{s_j} \quad (5)$$

Upon completion of the hierarchical clustering, the SC_q with the lowest value is searched for, which indicates that the partition P_q^* clusters are at a greater correlation level. The number of clusters at that moment should correspond to the actual number of devices. The aim of storing the partition at each time of the algorithm is to avoid rerunning the clustering because information of all the clusters in each iteration q is known. Next algorithm shows the proposal's pseudocode.

1. *Calculate $n^{(i)}$ of each image where $i \in 1, \dots, N$;*
2. *Generate the similarity matrix $H \in R^{N\times N}$;*
3. *Foreach $q \in 1, \dots, N-1$ do*
4. *Find cluster $H(k, l)$ with the highest similarity;*
5. *Remove the pair of rows and columns corresponding to clusters k and l;*
6. *Calculate the values of the new cluster using average link criteria and add the row and its corresponding column;*
7. *Determine the overall silhouette coefficient SC_q;*
8. *Store the partition P_q;*
9. *Find the partition where $min_q(SC_q)$.*

IV. Experiments and Results

The experiments were performed with a total set of 1050 photographs from 7 different mobile device camera models. The total set contains 150 photographs from each model. 7 devices are from different manufacturers (Apple iPhone 5, Huawei U8815, LG E400, Samsung GTS5830M, Zopo ZP980, Sony ST25a and Nokia 800 Lumia).

All the images were cropped to 1024×1024 pixels, all images have a horizontal orientation. The scenes of the photographs were chosen randomly, both indoors and outdoors, and they were also taken at different times and places in order to simulate a more realistic scenario. In the extraction of the noise pattern from all images, the zero - mean of rows and columns was used, 3 RGB color channels were converted to a single matrix in grayscale. Additionally, all experiments were conducted using the Wiener filter in the frequency domain.

To measure the degree of certainty in the results, the true positive rate TPR was used. The mean TPR for each of the following experiments is calculated, computing for each cluster the number of photos that have been well classified (TPR of each cluster) and averaging the TPRs of all the resulting clusters (if there are fewer clusters than devices the average takes into account the number of devices). To calculate the TPR of each cluster, the device that has the largest number of images with respect to the total of images by device needs to be identified within the cluster, that being the predominant device cluster, then calculate the percentage of photos that have been well classified for that device in the cluster. Actually, in the vast majority of cases it can be seen that a cluster is associated with one or more devices, as it can observed in matrices such as the ones in Tables I, II and III. If there are multiple clusters with the same number of photos from a device or a cluster with the same number of photos from several devices and in turn these being the highest, the cluster that is taken as predominant for the device is one chosen among the different options. It may be the case that if there is an extra cluster, a cluster may

doi:10.15849/icit.2015.0014

not be predominant for any device (see Table II) and its TPR for this cluster is 0. Or there might be one less cluster (see Table III), in this case the association of the cluster to a device will be taken into account and the number of devices will be used to calculate the average, as described above.

In Tables I, II and III there are examples that illustrate the calculation of the TPR for the three cases that may occur.

TABLE I. TPR WITH EQUAL NUMBER OF DEVICES THAN CLUSTERS

Brand - Model	Clusters (%)					Average TPR
	1	*2*	*3*	*4*	*5*	
Apple Iphone 5	49	0	0	1	0	
Huawei U8815	0	50	0	0	0	
LG E400	0	1	49	0	0	
Nokia 800 Lumia	0	0	0	50	0	
Samsung GT5830m	0	0	0	0	50	
TPR by cluster	**98**	**100**	**98**	**100**	**100**	**99.2 %**

In the results of the experiments 3 possible cases are considered: a) The number of identified clusters is equal to the number of devices, b) the number of identified clusters is higher than the number of devices, and c) the number of identified clusters is lower than the number of devices. Although the first case is ideal, in the second case classifications that do not mix different types of devices in a same cluster can be obtained.

TABLE II. TPR WITH LESS NUMBER OF DEVICES THAN CLUSTERS

Brand – Model	Clusters				Average TPR
	1	2	3	4	
Apple I-phone 5	100	0	0	0	
Huawei -U8815	0	100	0	0	
LG -E400	0	0	97	3	
TPR by cluster	**100**	**100**	**97**	**0**	**99 %**

TABLE III. TPR WITH MORE NUMBER OF DEVICES THAN CLUSTERS

Brand – Model	Clusters (%)				Average TPR
	1	2	3	4	
Apple Iphone 5	100	0	0	0	
Huawei U8815	0	100	0	0	
LG E400	0	0	100	0	
Nokia 800 Lumia	100	0	0	0	
Samsung GT 5830M	0	0	0	100	
TPR by cluster	**100**	**100**	**100**	**100**	**80 %**

V. CONCLUSIONS

This paper has made an analysis of the main unsupervised image grouping techniques, which are of utmost importance in digital image forensic analysis. Despite the rise of mobile device cameras these days, there are still few references for unsupervised mobile device image grouping in the state of the art. Most of the works refer to the supervised classification and in many cases they are not focused on mobile device images, which have unique characteristics. The noise added in every photograph by the camera sensor, due to the faults in its manufacturing process or defects from daily use, has proven to be a reliable source of device identification. Likewise, the calculation of normalized correlation between sensor noises extracted from two or more pictures is also a measure of similarity commonly used in unsupervised image learning techniques, clustering techniques being the ones which obtain the best results. The algorithm of this proposal is based on the combination of a hierarchical clustering and a flat clustering for the separation between clusters. The use of the silhouette coefficient for cluster validation proved to report good results when obtaining high TPRs; also, the number of clusters corresponded to the number of actual devices in most cases. Experiments conducted in this work have revealed a great diversity of situations with regard to the symmetry or not of the photo sets, their size, the number of devices used and the use of devices of the same brand. After all the experiments, it is concluded that the results of the application of the technique are good (92.7% TPR on average for all the experiments).

ACKNOWLEDGMENT

The research leading to these results has been partially funded by the European Union's H2020 Program under the project SELFNET (671672). Part of the computations of this work was performed in EOLO, the HPC of Climate Change of the International Campus of Excellence of Moncloa, funded by MECD and MICINN. This work was supported by the "Programa de Financiación de Grupos de Investigación UCM validados de la Universidad Complutense de Madrid – Banco Santander".

REFERENCES

[1] J. Hsu, "The Worldwide Mobile Phone Camera Module Market and Taiwan's Industry, 2010 and Beyond", 2010, pp. 1-18.

[2] R. Baer, "Resolution Limits in Digital Photography: The Looming End of the Pixel Wars - OSA Technical Digest (CD)", in *Proceedings of the Imaging Systems*, Tucson, Arizona United States, June 2010.

[3] T. Van Lanh, K.S. Chong, S. Emmanuel, M.S. Kankanhalli, "A Survey on Digital Camera Image Forensic Methods", in *Proceedings of the IEEE International Conference on Multimedia and Expo*, Beijing, pp. 16-19, July 2007.

[4] A. L. Sandoval Orozco, D. M. Arenas González, J. Rosales Corripio, L. J. García Villalba, J. C. Hernandez-Castro, "Techniques for Source Camera Identication", in *Proceedings of the 6th International Conference on Information Technology*, pp. 1-9, May 2013.

[5] J. Lukas, J. Fridrich, M. Goljan, "Digital Camera Identification from Sensor Pattern Noise", *IEEE Transactions on Information Forensics and Security*, IEEE, 2006, vol. 1 no. 2, pp. 205-214.

[6] C. F. Eick, N. Zeidat, Z. Zhao, "Supervised Clustering-Algorithms and Benefits", in *Proceedings of the IEEE International Conference on Tools with Artificial Intelligence*, Boca Raton, Florida, USA, pp. 774-776, November 2004.

[7] J. Fridrich, "Digital Image Forensics", *IEEE Signal Processing Magazine*, IEEE, 2009, vol. 26, no. 2, pp. 26-37.

[8] C.-T. Li, "Unsupervised Classification of Digital Images Using Enhanced Sensor Pattern Noise", in *Proceedings of the IEEE International Symposium on Circuits and Systems*, Paris, France, May 2010, pp. 3429-3432.

[9] L. Rokach, "A Survey of Clustering Algorithms", *Data Mining and Knowledge Discovery Handbook*, 2010, pp. 269-298.

[10] C. T. Zahn, "Graph-Theoretical Methods for Detecting and Describing Gestalt Clusters", *IEEE Transactions on Computers*, IEEE, 1971, vol. C-20, no. 1, pp. 68-86.

doi:10.15849/icit.2015.0014

[11] J. D. Banfield, A. E. Raftery, "Model-Based Gaussian and Non-Gaussian Clustering", *Biometrics*, Wiley, 1993, vol. 49, no. 3, pp. 803-821.

[12] D. H. Fisher, "Knowledge Acquisition Via Incremental Conceptual Clustering", *Machine Learning*, Springer, vol. 2. no. 2, pp. 139-172, 1987.

[13] J. Vesanto, E. Alhoniemi, "Clustering of the Self-Organizing Map", *IEEE Transactions on Neural Networks*, IEEE, 2000, vol. 11, no. 3, pp. 586-600.

[14] F. Hoppner, "Fuzzy Cluster Analysis: Methods for Classification", *Data Analysis and Image Recognition*, *Jossey-Bass Higher and Adult Education Series*, Wiley, 1999.

[15] S. Z. Selim, K. Alsultan, "A Simulated Annealing Algorithm for the Clustering Problem", *Pattern Recognition*, Elsevier, 1991, vol. 24, no. 10, pp. 1003-1008.

[16] B.-B. Liu, H.-K. Lee, Y. Hu, C.-H. Choi, "On Classification of Source Cameras: A Graph Based Approach", in *Proceedings of the IEEE International Workshop on Information Forensics and Security*, Seattle, Washington, USA, December 2010, pp. 1-5.

[17] R. Caldelli, I. Amerini, F. Picchioni, M. Innocenti, "Fast Image Clustering of Unknown Source Images", in *Proceedings of the IEEE International Workshop on Information Forensics and Security*, *Seattle, Washington*, December 2010, USA, pp. 1-5.

[18] M. Goljan, J. Fridrich, T. Filler, "Large Scale Test of Sensor Fingerprint Camera Identification", in *Proceedings of the Media Forensics and Security*, vol. 7254, San Jose, California, USA, 2009, pp. 72540I.

Al-Zaytoonah University of Jordan

The 7th International Conference on Information Technology

ICIT 2015
BigData

ISSN 2306-6105
ISBN 978-9957-8583-3-9

http://icit.zuj.edu.jo/

May 12th - 15th, 2015
Amman - Jordan

ICIT 2015 is Indexed by : IEEE EBSCO Google ULRICHSWEB GLOBAL SERIALS DIRECTORY IET Inspec

ICIT 2015 is Sponsored by :

ICIT 2015 The 7th International Conference on Information Technology
doi:10.15849/icit.2015.0105

New Technique of Forensic Analysis for Digital Cameras in Mobile Devices

Jocelin Rosales Corripio, Ana Lucila Sandoval Orozco, Luis Javier García Villalba
Group of Analysis, Security and Systems (GASS)
Department of Software Engineering and Artificial Intelligence (DISIA)
Faculty of Information Technology and Computer Science, Office 431
Universidad Complutense de Madrid (UCM)
Calle Profesor José García Santesmases, 9
Ciudad Universitaria, 28040 Madrid, Spain
Email: jocelinr@ucm.es, {asandoval, javiergv}@fdi.ucm.es

***Abstract*— Nowadays, forensic analysis of digital images is especially important, given the high use of digital cameras in mobile devices. The identification of the device type or the make and model of image source are two important branches of forensic analysis of digital images. In this paper we have addressed both, with an approach based on different types of image features and the classification using support vector machines. The study mainly has focused on images created with mobile devices and as a result, the techniques and features have been adapted or created for this purpose. There have been a total of 36 experiments classified into 5 sets, in order to test different configurations of the techniques. In the configuration of the experiments were taken into account among other things the future use of the technique by the forensic analyst in real situations and creating experiments with high technical requirements.**

Keywords*— Forensics Analysis, digital image, image source acquisition identification, image noise features, image color features, image quality metrics, image wavelet features

.

I. INTRODUCTION

Currently, the demand for mobile devices (mobile phones, smartphones, tablets, etc.) increases year by year despite the global economic crisis. According to Gartner [1] in 2013 smartphone sales grew 42.3% over the previous year, outnumber for the first time the sales of feature phones. We must not overlook the emergence in today's society of such devices in our day to day life. Increasing storage capacity, usability, portability and affordability, have allowed mobile devices to be present in several activities, places and events of daily life. A consequence of its widespread use, is that digital images can be used as silent witnesses in judicial proceedings (child pornography, industrial espionage, ...), and in many cases crucial pieces of an evidence of a crime [2].

Forensic analysis of digital images can be mainly divided into two branches [3]: tamper detection and image source identification. This work focuses on the first branch. Also, since mobile device cameras have some characteristics that make them different from the rest, this work focuses on images from this type of devices. In this paper, we propose a method to image source acquisition in mobile devices. The objective of this approach is to identify make and model from a group the different images into disjoint sets in which all their images belong to the same device. This paper is structured into 5 chapters, being the first this introduction. The rest of the paper is structured as follows. Section 2 shows carries out a state of the art of techniques and algorithms for identifying the source type and source acquisition identification. Section 3 shows different sets of features (Noise, Color, Image Quality Metrics (IQM) and Wavelets) used by the algorithms and techniques of forensic analysis. In section 4, a set of experiments for the identification of device type and the source acquisition identification of the image are performed. In these experiments we use the set of the features previously presented and the algorithms of the techniques. Finally, section 5 shows the main conclusions of this work and some future work lines.

II. RELATED WORK

The main techniques of digital image forensics for identifying the source of image acquisition and the main work of the analysis. The success of these techniques depends on the assumption that all the images acquired by the same device have intrinsic features. The features which are used to identify the make and model of a digital camera are derived from the differences between the techniques of image processing technologies and the components which are used. The biggest problem with this approach is that different models of digital cameras use components of a small number of manufacturers, and the algorithms used are also very similar between models of the same brand. According to [4] for this purpose four groups of techniques can be established depending on their base: lens system aberrations, Color Filter Array (CFA) interpolation, image characteristics, and sensor imperfections.

doi:10.15849/icit.2015.0105

Techniques Based on Image Features use a set of features extracted from the content of the image to identify the source. These features are divided into three groups: color features, Image Quality Metrics (IQM) and wavelet domain statistics.

[5] proposes a method to identify the source using the following features: color features, image quality metrics and frequency domain. The study adopted the wavelet transforms as a method to calculate the wavelet domain statistics and use a Support Vector Machine (SVM) for classification. In experiments digital cameras and mobile devices were used. The results obtained in different experiments show results between 61.7% and 99.72% accuracy.

In [6] authors extend the source identification to different devices such as mobiles, phones, digital cameras, scanners and computers. In this proposal they base it on the differences in the image acquisition process to create two features groups: color interpolation coefficients and noise features. In the experiments they use five smartphone models, five digital camera models and four scanner models to identify the source type. Their experiments showed an overall result of 93.75% accuracy. Identifying the maker and model of five mobile phone models resulted in an accuracy of 97.7%.

In [7] a method for source camera identification is proposed through the extraction and classification of wavelet statistical features. Finally 216 first-order wavelet features and 135 second order co-occurrence features is obtained. The most representative features are selected using an Sequential Forward Featured Selection (SFFS) algorithm and they are classified using a SVM. Identification success average of 98% the set of all cameras and an average success rate of 96.9% for the three cameras of the same model is achieved.

[13] performs experiments with common imaging features to identify the source: wavelet, color, IQM, statistical features of difference images and statistical features of prediction errors. In the experiments, different combinations of different types of features are used and a SVM for classification of different devices. Ten different cameras from four different makers with 300 images from each camera (150 for training and 150 for testing) and a resolution of 1024x1024 is used. Using all the features a score of 92% success rate is obtained. Moreover experiments were performed to check the robustness against three of the most common alterations in digital images: JPEG compression, cropping and scaling.

In [9] a technique for image source identification is proposed using ridgelets and contourlets subbands statistical models. After the feature extraction a SFFS algorithm is used for feature election and a SVM for classification. The method based on 216 wavelet features is considered useful only for the representation of a dimension, the approach based on ridgelets uses 48 features, and the approach based on contourlets includes a total of 768 features. In experiments with three cameras from different makers success rates are between 99.5% and 99.8%.

In [10] a method using the marginal density Discrete Cosine Transform (DCT) coefficients in low-frequency coordinates and neighboring joint density features from the DCT domain is proposed. Furthermore, hierarchical clustering and SVM is used to detect the source of acquisition of the images. In experiments with images from five smartphone models of four makers an accuracy of between 86.36% and 99.91% was obtained, achieving the best results with a linear SVM kernel.

I. PROPOSED WORK

Regarding classification, in [11] a study of different classification methods such as distance-based classifiers, Bayesian classifiers, neural networks, clustering algorithms and SVM classifiers is performed. As can be observed in the review, the use of SVM classifiers is widely used for these purposes. The kernel choice depends, among other factors, on the nature of the data to be classified. This paper will use an SVM classifier with Non-linear RBF kernel, as it is recommended for use when there is no a priori information about the data. The parameters for the SVM are the same as those used in [12]. Likewise, the option chosen is the most widely used one by the most recent precise works and they present good results. There are many implementations of SVM classifiers; particularly in this work we opted to use the LibSVM library [13].

The set of features to be used can be classified into four major groups, depending on the nature of their obtaining: noise features (16 features), color features (12 features), IQM (40 features) and wavelets (81 features). A detailed analysis on each of the aforementioned feature sets will be performed below.

A. Noise Features

One of the objectives is to get a set of features that allow us to differentiate between the different types of devices. To do this we firstly take into account that digital cameras use a two-dimensional array sensor whereas most scanners use a linear array sensor. In the case of scanners, the linear arrangement of the sensor moves to generate the entire image, so it is expected to find the periodicity of the sensor noise within the rows of the scanned image. On the other hand, there is no reason to find sensor noise periodicity within the columns of the scanned image. In the case of digital cameras this type of noise periodicity does not exist. This difference can be used as a basis to discriminate between different types of devices. Noise features extraction is based on [14].

Let I an image of $M \times N$ pixels, M as the rows and N as the columns. We denote I_{noise} the noise of the original image and I_{noise} is the image without noise.

$$I_{noise} = I - I_{denoise} \quad (1)$$

Then, each color component of the image without noise is subtracted to each color component of the original image, with which we obtain noise components of each pixel disaggregated for each color component.

The image original noise I_{noise} can be modeled as the sum of two components, the constant noise $I_{noiseconstant}$ and random noise $I_{noiserandom}$. For scanners constant noise only depends of the column index, because the same sensor is

doi:10.15849/icit.2015.0105

moved vertically to generate the complete image. The average noise of all columns can be used as a pattern reference $I_{noiseconstant}(1,j)$ because the random noise components were cancelled. For detecting the similarity between different rows with the pattern reference, we use the correlation of these rows with the pattern.

$$corr(X,Y) = \frac{(X-\overline{X})\,(Y-\overline{Y})}{\|X-\overline{X}\|\,\|Y-\overline{Y}\|} \quad (2)$$

Then the same process is performed to detect the similarity of the columns with the pattern reference. After obtaining the correlation between rows and between columns we will go to obtain the feature set. It should be noted at the time of obtaining the features, that in the case of scanners the orientation of the image is critical, because features obtained will be completely different.

For each type of correlation first order statistical values are obtained, which are: mean, median, maximum and minimum. Also, the ratio features between rows and columns correlations are added. Finally the average noise per pixel feature was included. This feature does not depend on rows or columns correlations with the reference pattern, but is independent and it can distinguish between different types of devices, such as computer generated images. In total a set of 16 features are obtained: 7 rows features, 7 columns features, the ratio between rows and columns correlations and the average noise per pixel.

B. Color Features

The configuration of the CFA filters, the demosaicing algorithm and color processing techniques mean that signals in the color bands may contain treatments and specific patterns. In order to determine the differences in color features for different camera models, it is necessary to examine the first and second order statistics of the pictures taken with them.

- *Pixels average value*: This measure is performed for each RGB channels (3 features).
- *Correlation pair between RGB bands*: This measure expresses the fact that depending on the structure of the camera, the correlation between the different color bands can change (3 features which come from measuring the correlation between the RG, RB and GB bands).
- *Neighbor distribution center of mass for each color band*: This measure is calculated for each band separately (3 features). Firstly, the total number of pixels for each color value is calculated, obtaining a vector with 256 components. Then, with these calculated values the sum of neighboring values are obtained.
- *Energy ratios between pairs RGB*: This feature depends on the white dots correction process of the camera (3 features)

C. Image Quality Metrics

Different camera models produce images of different quality. There may be differences in image brightness, sharpness or quality color. These differences propose a set of quality metrics features that help us to distinguish the image source. There are different IQM categories: measures based on the pixels differences, measures based on correlation and measures based on spectral distance. For obtaining this set of metrics, a filtered image in which the noise of the original image is reduced to perform different calculations is needed in addition to the original image. For this, a Gaussian filter that allows us to perform image smoothing is used. After the core is obtained, it is normalized, so that the sum of all its components is 1. This is necessary to obtain a smooth image but with the same colors as the original. The normalization is performed dividing each component by the sum of the values of all the components. For obtaining the metrics a filter with a 3x3 kernel with $\gamma = 0.5$ is used. Following the specification of the 40 IQM features based on [8].

- *Czekonowsky distance*: The Czekonowsky distance is a useful metric for comparing vectors with no negative components as in the case of color images.
- *Minkowsky metrics*: Minkowsky metrics for $\gamma = 1$ and $\gamma = 2$.
- *Normalized Cross Correlation*: The closeness between two digital images can also be quantified in terms of a correlation function. The quality metric of the normalized cross-correlation measurement for each image band *k*.
- *Structural Content*: The structural content of an image quality metric is defined for each band *k*.
- *Spectral Measures*: The Discrete Fourier Transform (DFT) of the original image and the smoothed image, denoted as $\tau_k(u,v)$and $\hat{\tau}_k(u,v)$ for a band *k*.
- *Measures based on the human visual system*: Images can be processed by filters which simulate the Human Visual System (HVS). One of the models used for this is a band-pass filter with a transference function in polar coordinates.

D. Wavelet Features

Due to the deterministic property of the sensor pattern noise which is present in an image, this pattern can be used as a footprint to identify the device that generated the image under investigation. It can be said that the sensor pattern noise is to a digital camera as a fingerprint is to a human being. To identify the acquisition source we require an algorithm that allows us to extract the sensor noise and another that allows us to obtain the features of the fingerprints obtained in order to classify and identify them.

Taking the main ideas from [15] as a reference, algorithm 1 is proposed to extract sensor noise.

Algorithm 1: Extracting PRNU

```
1.  Apply a wavelet decomposition in 4 levels to I;
2.      Foreach wavelet decomposition level do
3.          Foreach component c ϵ {H,V,D} do
4.              Compute the local variance;
5.              If (adaptive variance)
6.                  Compute 4 variances with windows of size: 3,
                    5, 7 and 9 respectively;
7.                  Select the minimum variance;
8.              else
9.                  Compute the variance with a window of size 3;
10.             Compute noiseless wavelet components applying
                the Wiener filter to the variance;
```

doi:10.15849/icit.2015.0105

11. *Obtain I_{clean} by applying the inverse wavelet transform with clean components calculated;*
12. *Obtain the sensor noise with I_{noise}=I- I_{clean};*
13. *Apply zero-meaning to I_{noise};*
14. *Increase the green channel weight with*
$$I_{noise} = 0.3 \cdot I_{noise_R} + 0.6 \cdot I_{noise_G} + 0.3 \cdot I_{noise_B};$$

Finally, a total of 81 features (3 channels x 3 wavelet components x 9 central moments) are calculated using algorithm 2.

Algorithm 2: Extracting features

1. *Separate R, G and B color channels of I_{noise};*
2. ***Foreach** color channel **do***
3. *Apply a wavelet decomposition in 1 level;*
4. ***Foreach** component c ϵ {H,V,D} **do***
5. *Compute k central moments with*
$$m_k = \frac{1}{n}\sum_{i=1}^{n}|c - \bar{c}|^k$$

II. Experiments and Results

We performed the classification of images on closed set of elements, i.e., the classes of the elements used in training are the same classes as those used in the test. The images used in the training stage are not used in the testing stage.

In order to evaluate the source device type identification we will use an image set composed of: images from mobile phones, images obtained from a scanner, and a computer-generated images. 200 images are used from each set, 100 for the SVM training and 100 for testing. All images have a resolution higher than 1024x768. There is no restriction on the content of the image or the camera configuration parameters at the time of the acquisition.

Images from 7 smartphones: IPhone 4s (I1), Blackberry 8520 (BB), Huawei U8815 (HU), LG P760 (LG2), Nokia 800 (N1), Samsung GT-I9001 (S1) and Sony C2105 (SE1). For images from scanners and computer-generated images, our own sources and the *Flickr* website were used. As a second filter for scanned images, those which had the tag "scanned images" and made reference to a retail scanner model were used. For the experiments we have taken into account the following configuration parameters: size of crop applied to the image, crop position (centered or upper-left corner) and application of different feature sets (Noise Features, Color Features, IQM Features and Wavelet Features).

Table I shows the results of success rates to evaluate the source device type identification between Camera (A), Computer (B) and Scanner (C), and the configuration parameters used in the 10 experiments.

From the analysis of the results, general and specific conclusions about the various configurations used in each experiment can be obtained. Encompassing all the experiments, it is observed that success rates are not excessively high (60.42% on average and 71.30% in the best case); it can be concluded that this technique is not particularly suitable for this purpose. It is important to emphasize, as noted above, that the number of different makes and models used for this experiment is high, which predictably causes success rates to drop. That being said, it should be noted that this study does provide interesting results on the configuration parameters used, since between the best and the worst result there is a difference in the average success rate of 23.48%.

TABLE I. TPR WITH EQUAL NUMBER OF DEVICES THAN CLUSTERS

Features	Crop Size	Crop Align	Device (%)			Average (%)
			A	B	C	
Noise	Full Size	-	70	54	57	59.95
	1024x768	Center	66	80	46	62.39
	800x600		76	60	49	60.68
	640x480		62	61	48	56.62
	1024x768	Upper-left corner	76	59	40	56.40
	800x600		65	38	44	47.72
	640x480		74	54	37	52.88
All Features	1024x768	Center	66	73	72	70.26
	800x600		69	74	71	71.30
	640x480		77	73	63	70.75
Average			**69.9**	**61.3**	**51.4**	**60.42**

Given the importance of mobile images today, below we will show the experiment performed to identify the acquisition source of images from mobile devices, i.e., the classification of an image set according to the make and model of the camera that generated them.

The results improve significantly when all the features to identify the source type are used. Given the high number of classes, the results can be qualified as acceptable, since the average success rate for all experiments carried out using these features is 70.77%. The experiments have been grouped into 3 groups with the aim of obtaining conclusions on: the use of different feature sets, crop size, the number of devices used for the classification, and the use of devices from the same manufacturer.

Table II shows the experiments in which 7 models of mobile devices from different manufacturers are used. Different types of combinations of features sets were tested. Most experiments were performed with a crop size of 1024x768, since as this is considered a large enough size to obtain good results, as shown in the previous experiments.

TABLE II. TPR WITH EQUAL NUMBER OF DEVICES THAN CLUSTERS

Features	Crop Size	Crop Align	I1	HU	LG2	N1	BB	S1	SE1	Average
All Features (Daubechies 8-tap)	1024x768	Center	93	96	80	94	91	70	85	86.54

doi:10.15849/icit.2015.0105

Noise	1024x768	Center	41	42	35	18	40	40	62	37.67
Color	1024x768	Center	24	37	20	40	31	19	44	29.27
IQM	1024x768	Center	13	88	46	89	7	34	2	21.65
Wavelet Daubechies 8-tap	1024x768	Center	95	96	96	94	92	76	93	91.46
Wavelet Haar	1024x768	Center	95	87	97	70	86	56	91	81.84
Color + IQM + Wavelet Daubechies 8-tap	1024x768	Center	93	94	90	90	90	53	85	83.67
All Features (Daubechies 8-tap)	800x600	Center	91	96	84	92	95	56	85	84.41
All Features (Daubechies 8-tap)	640x480	Center	90	95	84	89	88	51	88	82.15

The experiment reveals that noise, color and IQM feature sets are individually completely invalid, since the best result obtains an 37.67% average success rate, which is unacceptable. With the remaining set of features (wavelets), two experiments were conducted using different types of wavelet: Daubechies 8-tap and Haar. The results show that Daubechies 8-tap obtains better results than Haar and the best results of all experiments (91.46%).

With respect to the different feature combinations, it is observed that when we use all the features good results are obtained (86.54% in the best case), since, although they are slightly worse than the best result, the difference is not very significant (4.92%). Also, the success rate when all the features are used subtly drops the smaller the crop size gets.

The combination of all the features except noise features, which are mainly focused on identifying the source type, yields an average success rate of 83.67%. These results, even if not bad, are far from those obtained with the wavelets and worse than when the combination of all features is used.

CONCLUSIONS

In this work we have presented various techniques for identifying mobile device images with respect to scanned and computer-generated images. Besides, other techniques that allow us to distinguish the acquisition source of smartphone images are presented. The techniques are based on the use of four feature sets (Noise, Color, IQM and Wavelets), on which adjustments have been made in order to improve the results for this specific type of devices. There have been experiments with the combination of the different feature sets, different crop sizes and positions, and wavelet functions. With regard to source type identification, the first general conclusion is that Noise features are discarded as invalid when the number of types of devices is greater than 2. In the experiments that used whole images and different crop sizes and positions, unacceptable results were obtained for identifying three types of devices (scanner, smartphone and computer). As discussed in the experiments, for these three types of devices there are dozens of different manufacturers and models, hampering classification. As a counterpart, forensic analysts may consider the application of the technique with Noise features for identifying the source type of images from mobile devices with respect to images from scanners and computers. The results are quite good at identifying the type when discerning between scanners and smartphones. The use of all the features significantly improves results, but as a general conclusion they are not good enough to be used in a serious situation. When identifying the acquisition source of mobile device images, the results are much more encouraging. In all sets of experiments performed, there is at least one configuration that yields good results, always putting them into the context of the level of demand on this technique (a large number of devices or many devices from the same manufacturer).

ACKNOWLEDGMENT

The research leading to these results has been partially funded by the European Union's H2020 Program under the project SELFNET (671672). Part of the computations of this work was performed in EOLO, the HPC of Climate Change of the International Campus of Excellence of Moncloa, funded by MECD and MICINN. This work was supported by the "Programa de Financiación de Grupos de Investigación UCM validados de la Universidad Complutense de Madrid – Banco Santander".

REFERENCES

[1] Gartner Says Smartphone Sales Grew 46.5 Percent in Second Quarter of 2013 and Exceeded Feature Phone Sales for First Time (2013). URL http://www.gartner.com/newsroom/id/2665715

[2] M. Al-Zarouni, "Mobile Handset Forensic Evidence: a Challenge for Law Enforcement", in *Proceedings of the 4th Australian Digital Forensics Conference*, School of Computer and Information Science, Edith Cowan University, 2006.

[3] T. Gloe, M. Kirchner, A. Winkler, R. Bohme, "Can We Trust Digital Image Forensics?", in *Proceedings of the 15th International Conference on Multimedia*, ACM Press, 2007, pp. 78–86.

[4] T. Van Lanh, K. S. Chong, S. Emmanuel, M. S. Kankanhalli, "A Survey on Digital Camera Image Forensic Methods", *IEEE International Conference on Multimedia and Expo*, IEEE, 2007, pp. 16–19.

[5] M. J. Tsai, C. L. Lai, J. Liu, "Camera/Mobile Phone Source Identification for Digital Forensics",in *Proceedings of the International Conference on Acoustics Speech and Signal Processing*, IEEE, 2007, pp. II–221–224.

[6] C. Mckay, A. Swaminathan, H. Gou, M. Wu, "Image Acquisition Forensics: Forensic Analysis to Identify Imaging Source", in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, IEEE, 2008, pp. 1657–1660.

[7] B. Wang, Y. Guo, X. Kong, F. Meng, "Source Camera Identification Forensics Based on Wavelet Features", in *Proceedings of the International Conference on Intelligent Information Hiding and Multimedia Signal Processing*, IEEE Computer Society, 2009, vol. 0, pp. 702–705.

[8] Y. Hu, C. T. Li, C. Zhou, "Selecting Forensic Features for Robust Source Camera Identification", *Computer Symposium (ICS)*, 2010 International, 2010, pp. 506–511.

[9] L. Ozparlak, I. Avcibas, "Differentiating Between Images UsingWavelet-Based Transforms: A Comparative Study", *IEEE Transactions on Information Forensics and Security*, IEEE, 2011, vol. 6, no. 4, pp. 1418–1431.

doi:10.15849/icit.2015.0105

[10] Q. Liu, X. Li, L. Chen, H. Cho, A. P. Cooper, Z. Chen, M. Qiao, A. H. Sung, "Identification of Smartphone-Image Source and Manipulation", *Advanced Research in Applied Artificial Intelligence, Lecture Notes in Computer Science*, Springer Berlin Heidelberg, Dalian, China, 2012, vol. 7345, pp. 262–271.

[11] D. Michie, D. J. Spiegelhalter, C. C. Taylor, "Machine Learning, Neural and Statistical Classification", *Ellis Horwood*, 1994.

[12] J. Rosales Corripio, D. M. Arenas González, A. L. Sandoval Orozco, L. J. García Villalba, J. C. Hernandez-Castro, S. J. Gibson, "Source Smartphone Identification Using Sensor Pattern Noise and Wavelet Transform", In *Proceedings of the 5th International Conference on Imaging for Crime Detection and Prevention (ICDP 2013)*, pp. 1–6 2013.

[13] C. C. Chang, C. J. Lin, "LIBSVM: A Library for Support Vector Machines". Version 3 17, Abril 26, 2013. URL http://www.csie.ntu.edu.tw/˜cjlin/libsvm/

[14] N. Khanna, A. K. Mikkilineni, E. J. Delp, "Scanner Identification Using Feature-based Processing and Analysis", *IEEE Transactions on Information Forensics and Security*, IEEE, 2009, vol. 4, no. 1, pp. 123–139.

[15] J. Lukas, J. Fridrich, M. Goljan, "Digital Camera Identification from Sensor Pattern Noise", *IEEE Transactions on Information Forensics and Security*, IEEE, 2006, vol. 1, no. 2, pp. 205–214,.

Al-Zaytoonah University of Jordan

The 7th International Conference on Information Technology

ICIT 2015
BigData

ISSN 2306-6105
ISBN 978-9957-8583-3-9

http://icit.zuj.edu.jo/

May 12th - 15th, 2015
Amman - Jordan

ICIT 2015 is Indexed by : IEEE EBSCO Google ULRICHSWEB GLOBAL SERIALS DIRECTORY IET Inspec

ICIT 2015 is Sponsored by :

IET Computer Vision

Research Article

IET Journals
The Institution of Engineering and Technology

Smartphone image acquisition forensics using sensor fingerprint

ISSN 1751 9632
Received on 1st August 2014
Revised on 26th October 2014
Accepted on 24th November 2014
doi: 10.1049/iet cvi.2014.0243
www.ietdl.org

Ana Lucila Sandoval Orozco[1] ✉, Luis Javier García Villalba[1], David Manuel Arenas González[1], Jocelin Rosales Corripio[1], Julio Hernandez-Castro[2], Stuart James Gibson[3]

[1]*Group of Analysis, Security and Systems (GASS), Department of Software Engineering and Artificial Intelligence (DISIA), School of Computer Science, Office 431, Universidad Complutense de Madrid (UCM), Calle Profesor José García Santesmases s/n, Ciudad Universitaria, 28040 Madrid, Spain*
[2]*School of Computing, University of Kent, Canterbury CT2 7NF, UK*
[3]*School of Physical Sciences, University of Kent, Canterbury, Kent, CT2 7NH, UK*
✉ *E mail: asandoval@fdi.ucm.es*

Abstract: The forensic analysis of digital images from mobile devices is particularly important given their quick expansion and everyday use in the society. A further consequence of digital images' widespread use is that they are used today as silent witnesses in legal proceedings, as crucial evidence of the crime. This study specifically addresses the description of a technique that allows the identification of the image source acquisition, for the specific case of mobile devices images. This approach is to extract wavelet-based features from sensor pattern noise which are then classified using a support vector machine. Moreover, there are a number of parameters that allows the authors to adapt the execution of the algorithm to specific situations desired for the forensic analyst (a variety of types and sizes of image or optimising the average accuracy rate in terms of processing time). This article describes a set of experiments with the same set of images that can obtain general conclusions for the different configurations.

1 Introduction

Owing to increasing storage capacity, usability, portability and affordability, camera enabled mobile phones have become ubiquitous consumer electronic devices. The development of digital technologies has been advancing and continues to do so at an unstoppable rate. Every day the number of digital cameras is growing as well as the ease of access to them. Mobile digital cameras deserve special attention. According to Gartner [1], 1.745 billion handsets were sold in 2012 and it is predicted that 1.9 billion handsets will be sold in 2013. In total, according to estimates by the International Communication Union, there are 6.8 billion mobile phone subscriptions worldwide, which is a large increase from the 6 billion subscriptions in 2012 and 5.8 billion in 2011.

83% of these mobile devices have an integrated digital camera, which in contrast to conventional digital cameras are carried by their owners all the time to most places they attend and, in many cases, these devices have internet access [2]. The quality of these cameras has increased so much that many people use them as a replacement for digital still cameras (DSCs). In 2012, 31% of digital cameras sold belong to mobile phones, PCs and tablets and the forecast for 2016 according to [3] is to increase to 48%. In 2013 only 27% of market share will be from DSCs. There are also predictions that DSCs will disappear in favour of new integrated mobile device cameras [4], because the improved quality of these devices is growing at an unstoppable rate.

Having described this overview in figures on the extent of the presence of mobile devices in the world, we must not overlook the emergence in today's society of such devices in our day to day life. So much so, that according to Ahonen *et al.*, [2], a large number of people have and use more than one mobile device and a typical user turns to their mobile devices an average of 150 times a day.

The extensive use of smartphone cameras makes enforcing legal restrictions on the capture and sharing of digital photographs very difficult. Restrictions on the use of cameras include locations such as schools, government offices and businesses. Consequently, tools which permit the identification of source devices have significant utility in various areas of law enforcement [5] such as child protection or digital rights management.

Often the pictures are considered to be real events captured by digital cameras. However, with the development of technology, powerful and sophisticated tools have emerged that facilitate the alteration of digital images in an impressive manner, even for those without technical knowledge or expertise in the area [6].

For these reasons, nowadays, digital image forensic analysis of mobile devices is very important. The study should be specific to mobile device images, because they have specific characteristics that allow for better results, not as valid digital image forensic techniques but for other kind of devices.

2 Image acquisition process in a digital camera

The first step to understanding and creating image forensic algorithms is to know in detail the image acquisition process in digital cameras. This process is summarised in Fig. 1.

Although many details of the camera pipeline belong to each manufacturer, the general structure is the same in all of them. Below is a brief description of each image acquisition phase.

When capturing an image, it is necessary to measure three or more bands for each pixel, which requires more than one sensor, and consequently it increases the cost of the camera. The most widespread and economical solution is the placement of a 'colour filter array' (CFA) in front of the sensor. There may be mechanisms interacting with the sensor to determine the exposure (aperture size, shutter speed and automatic gain control) and the focal length of the lens.

An antialiasing filter is also placed before the sensor; this filter is in charge of cleaning the signal prior to the analogue to digital conversion. This filter generates smoother contours in the image, reducing the unpleasant staggered appearance of lines.

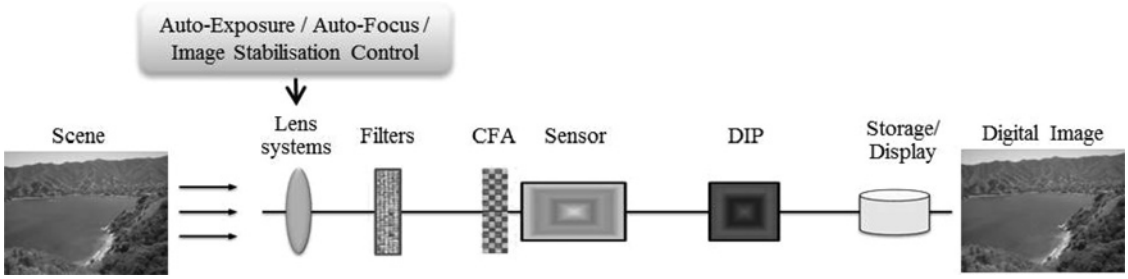


Fig. 1 *Image acquisition process in a digital camera*

The sensor (charge coupled device (CCD) or complementary metal oxide semiconductor) records the image converting light energy into electrical energy. The raw data obtained from the sensor needs to be processed to remove noise and other artefacts (anomalies introduced into digital signals). One of these processes is the correction of defective pixels caused by imperfections in the sensor, which corrects these pixels by interpolation. Another process is the white balance that allows for a more accurate colour reproduction without dominant colours; this effect is especially noticeable in neutral colours such as white. Demosaicing is the most complex process from the computational point of view and the techniques used are often owned by the camera manufacturer. This algorithm uses the values of the neighbouring pixels to calculate the values of the channels that have not been measured (remember that each pixel sensor detects only the channel that the array CFA allows to pass).

Another process to which the image is subjected is called gamma correction, which adjusts the intensity values of the image. Although these algorithms are in the pipeline from any camera, the exact process may vary from one manufacturer to another, and even from one camera model to another. Finally the image is compressed (mobile phone cameras typically use the algorithm joint photographic experts group (JPEG)) to save space. The compressed image is stored in the device memory with the image information in EXIF [7].

In [8] the image acquisition process in cameras of mobile devices is described, likewise a comparison of this process compared with that in DSCs and scanners is presented.

3 Source camera identification techniques

Research in this field studies the design of techniques to identify maker and model of the devices used to generate digital images. Analogously to ballistic analysis trying to relate a gun with its bullets, digital image forensics tries to identify the link between images and the digital camera which has generated them [9]. The success of these techniques depends on the assumption that the characteristics are unique to each device. The characteristics used to identify the maker and the model of digital cameras are derived from the differences between image processing techniques and technologies used in camera components [10]. The main problem with this approach is that different models of digital camera are often built using the same core components that originate from a small number of manufacturers. As a consequence it can be difficult, or in some cases impossible, to differentiate between models using such methods.

According to Van Lanh *et al*. [10], for this purpose four groups of techniques can be established depending on their base: lens system aberrations, CFA interpolation, image characteristics and sensor imperfections. The latter is the subject of this paper. In addition to the above there is another group of techniques based on metadata.

Metadata techniques are the simplest and there is plenty of research based on them. However, these techniques are highly dependent on the metadata that manufacturers decide to insert when generating images. Moreover, this method is the most vulnerable to malicious modifications or even the total elimination of metadata either intentionally or unwittingly.

During the image generation process the lens system can introduce some aberrations (spherical, coma, astigmatism, field curvature, radial distortion and chromatic aberration). The radial distortion is the one with the most impact over pictures, especially in cameras having cheap wide angle lenses. Most digital cameras use this type of lens for cost reasons. In [11] the lens radial distortion is proposed as the best technique for source identification. Radial distortion causes straight lines to appear as curves in images. The radial distortion degree of each image can be measured by a process consisting of three steps: edge detection, distorted segment extraction, and distortion error measurement. Choi experimented with three different cameras and obtained 91.28% accuracy identifying the camera source.

Some authors consider that CFA choice and the interpolation algorithm specifications generate some of the most striking differences between different camera models.

In [12] an algorithm for identifying and classifying colour interpolation operations is presented. This proposal is based on two methods to perform the classification process: first using an algorithm to analyse the correlation of each pixel value with its neighbours' values, and secondly an analysis of the differences between pixels independently. The accuracy for the source camera identification with images from four to five different models were of 88% and 84.8%, respectively.

In [13] correlations between pixels are used for the source identification, obtaining a coefficient matrix for each colour channel while defining a pixel quadratic correlation model. Neutral networks are used for classification. The method was tested with cartoon images from four cameras. The success rate obtained was 98.6%. This approach is not efficient at differentiating between different models from the same maker.

In [14] a set of binary similarity measures is used as metrics to estimate the similarity between image bit planes. The fundamental assumption of this work is that CFA interpolation algorithms from each maker leave correlations along image bit planes and can be represented by a set of 108 binary similarity measures for classification. The success rate of their experiments was between 81 and 98% to classify three cameras and decreased to 62% to identify between nine cameras.

The techniques based on image features use a set of features extracted from image content to identify the source. These features are divided into three groups: colour characteristics, image quality metrics and wavelet domain statistics.

In [8], the authors extend the source identification to different devices such as mobiles, phones, digital cameras, scanners and computers. In this proposal, colour interpolation coefficients and noise characteristics are used to classify. Their experiments showed an overall result of 93.75% accuracy. Identifying the maker and model of five mobile phone models, the accuracy obtained was 97.7%.

In [15], a method based on the bi coherence statistics phases and magnitudes along with the wavelet coefficients is used for the identification. This method captures the unique nonlinear distortions in the wavelet domain produced by the cameras when performing processing operations over images. As a result an accuracy of 97% in the identification was obtained in distinguishing different models from the same manufacturer.

In [16], a technique to differentiate images using the wavelet family transforms is explained. Ridgelets and contourlets subbands

IET Comput. Vis., 2015, Vol. 9, Iss. 5, pp. 723 731

statistical models are proposed to extract the representative features from images. Experiments were conducted to identify three different cameras obtaining accuracies of 93.3% with wavelet based approach, 96.7% using ridgelets, and 99.7% with contourlets.

In [17], a method using the marginal density discrete cosine transform (DCT) coefficients in low frequency coordinates and neighbouring joint density features on both intra block and inter block from the DCT domain is proposed. In experiments with images of different scale factors from five smartphone models of four makers, an accuracy of between 86.36% and 99.91% was obtained.

The techniques based on sensor noise study the traces left by sensor defects in images. These techniques are mainly divided into two branches: pixel defects and sensor pattern noise (SPN). The first branch studies pixel defects, hot pixels, dead pixels, row or column defects, and group defects. In the second branch a pattern is constructed by averaging multiple residual noises computed by any noise removal filter; The presence of the pattern is determined using a correlation method or machine classification support vector machine (SVM).

In [18], pixel defects of CCD sensors are studied, focusing on different features to analyse images and then identify their source: CCD sensor defects, the file format used, noise introduced in the image and watermarking introduced by makers. Among the CCD sensor defects are considered hot spots, dead pixels, group defects, and row/column defects. Results indicate that each camera has a different defect pattern. Nevertheless, it is also noted that the number of pixel defects for images from the same camera is different and varies greatly depending in the image content. Likewise, it was revealed that the number of defects varies with temperature. Finally, the study found that high quality CCD cameras do not have this kind of problem. When considering only defective CCD sensors this study is not applicable to the analysis of images generated by mobile devices.

In [19], the authors analyse the SPN from a set of cameras, which functions as a fingerprint allowing the unique identification of each camera. This pattern noise is obtained by averaging the sensor noise extracted from different images with a noise removal filter. To identify the camera from a given image, the reference pattern is considered as a watermark in the image and its presence is established by a correlation detector. It was found that this method is affected by processing algorithms such as image JPEG compression and gamma correction. The results for pictures with different sizes were unsatisfactory [10].

In [20], an approach to source camera identification in open set scenarios is proposed, where unlike closed scenarios it is not assumed to have access to all the possible image source cameras. This approach, in contrast to others, considers nine different regions of interest (ROIs) located in the corners and the centre of the images (not only the central region of the image). Using these ROIs, it is possible to work with different resolution images without requiring zero padding or colour interpolating. The SPN is computed for each colour channel generating a total of 36 representative features for each image. Then, the features of images taken by the camera under investigation are labelled as positive class and features from images made by other cameras as negative classes. After the SVM training phase, in which the hyper plane that separates the positive and negative classes is estimated, this hyperplane is moved by a given value either inward (for positive classes) or outward (for negative classes) for the purpose of considering the open scenario unknown classes. The results had an accuracy of 94.49, 96.77 and 98.10%.

In [21], the sensor noise is extracted by calculating similarities as a classification method on the basis of [19]. The authors state that the sensor noise can be highly contaminated by the scenario details, and they propose that the stronger a component of the sensor noise is, the less reliable it is and therefore it should be attenuated. They performed experiments with six different DSCs. For images of 1536×2048 pixels, they obtained an accuracy of 38.5% with the implementation without the improvement and 80.8% with the proposed improvement. For images of 512×512 pixels, they obtained an accuracy of 21.8% without improvement and 78.7% with the proposed improvement.

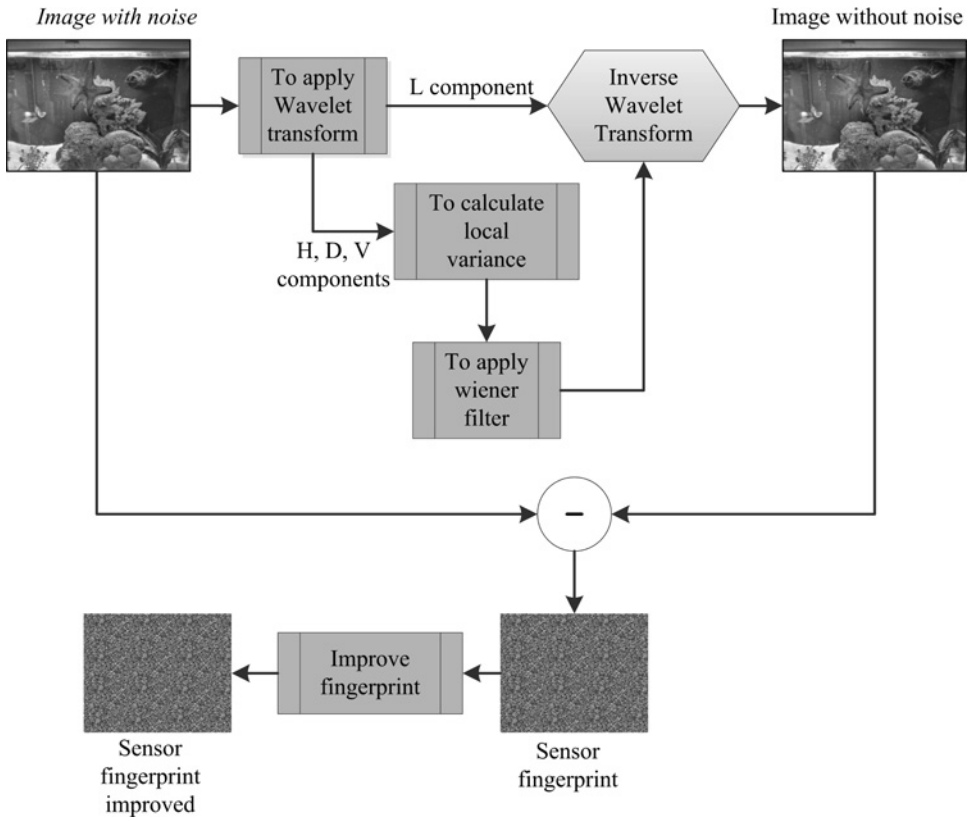


Fig. 2 *Scheme functional*

A detailed comparison of different source identification techniques is presented in [22].

4 Source identification algorithm

Previous work has shown SPN [18, 22, 19] and wavelet transform [15, 16] to be an effective method for source camera identification. However, almost all studies have focused only on traditional cameras, excluding mobile cameras. This makes it an area of study that requires attention especially with mobile devices. Using a biometric analogy, we consider each noise pattern to be a fingerprint of its source camera's sensor. In our study, SPN is used to classify images captured by camera enabled smartphones. Our approach characterises the fingerprints using wavelet based feature vectors. The scheme presented in Fig. 2 shows the functional diagram of our proposal.

Noise images were obtained using the method previously described by [19] and also summarised by Fig. 3 as follows.

To extract its noise pattern, an image is decomposed into its red, green and blue colour channels. Then, a four level wavelet decomposition of each colour channel is calculated using the Daubechies, 8 tap, separable quadrate mirror filters. The number of decomposition levels can be increased to improve accuracy or to reduce processing time.

Horizontal H, vertical V and diagonal D high frequency images are obtained for each level of decomposition. For each detail image, the local scene variance in a $W \times W$ window is estimated. Four estimates are obtained with window sizes corresponding to $W \in \{3, 5, 7, 9\}$. Finally, we choose the estimate which maximises the *a posteriori* probability

$$\hat{\sigma}^2(i,j) \quad \max\left(0, \frac{1}{W^2}\sum_{(i,j)\in N} c^2(i,j) \quad \sigma_0^2\right), \quad (i,j) \in J \qquad (1)$$

Algorithm 1:

Input: Image I
Variance estimation: adaptive or non-adaptive
Result: Sensor fingerprint I_{noise}

1 **procedure** EXTRACTPRNU(I)
2 Apply a wavelet decomposition in 4 levels to I;
3 **foreach** *wavelet decomposition level* **do**
4 **foreach** *component* $c \in \{H, V, D\}$ **do**
5 Compute the local variance;
6 **if** *adaptive variance* **then**
7 Compute 4 variances with windows of size: 3, 5, 7 and 9 respectively;
8 Select the minimum variance;
else
9 Compute the variance with a window of size 3;
10 Compute noiseless wavelet components applying the Wiener filter to the variance;
11 Obtain I_{clean} by applying the inverse wavelet transform with clean components calculated;
12 Obtain the sensor noise with $I_{noise} = I - I_{clean}$;
13 Apply zero-meaning to I_{noise};
14 Increase the green channel weight with $I_{noise} = 0.3 \cdot I_{noise_R} + 0.6 \cdot I_{noise_G} + 0.1 \cdot I_{noise_B}$;
15 **end procedure**

Fig. 3 *Extracting PRNU*

Algorithm 2:

Input: Sensor fingerprint I_{noise}
Result: 81 features

1 **procedure** EXTRACTFEATURES(I)
2 Separate R, G and B color channels of I_{noise};
3 **foreach** *color channel* **do**
4 Apply a wavelet decomposition in 1 level;
5 **foreach** *component* $c \in \{H, V, D\}$ **do**
6 Compute k central moments with $m_k = \frac{1}{n}\sum_{i=1}^{n} |c_i - \bar{c}|^k$;
7 **end procedure**

Fig. 4 *Extracting features*

where, $c(i,j)$ is the high frequency component and $c \in \{H, V, D\}$; σ_0 controls the degree of noise suppression.

The minimum of four variances is chosen as the best estimate

$$\hat{\sigma}^2(i,j) \quad \min\left(\begin{matrix}\sigma_3^2(i,j), \sigma_5^2(i,j),\\ \sigma_7^2(i,j), \sigma_9^2(i,j)\end{matrix}\right), \quad (i,j) \in J \qquad (2)$$

An alternative and less accurate method is to simply use $W \quad 3$ as the estimated local variance.

The denoised wavelet coefficients are defined by the Wiener filter as follows

$$c_{\text{clean}}(i,j) \quad c(i,j)\frac{\hat{\sigma}^2(i,j)}{\hat{\sigma}^2(i,j) + \sigma_0^2} \qquad (3)$$

The noise residual is obtained by calculating the inverse transform and subtracting the denoised image from the original image. JPEG and demosaicing artefacts, presented in the noise image, are suppressed by subtracting the mean column and row values [23]. Greater weight is given to the green channel since the configuration of the colour matrix this channel contains more information about the image [24 26].

The next step is to obtain features that characterise the sensor fingerprint for the purpose of classification. A total of 81 features (3 channels × 3 wavelet components × 9 central moments) is extracted using the Fig. 4.

Classification was performed using a SVM with RBF kernel. We used the LibSVM package in which the SVM is extended to multiple classes yielding class probability estimates [27]. A grid search was used to obtain the best kernel parameters (γ and C). The classifier was trained and tested with feature vectors extracted from randomly selected images.

5 Experiments and results

To assess the effectiveness of the proposed algorithms, a set of experiments have been made with a variety of configuration parameters. Table 1 summarises the parameters used and their possible values.

The PRNU extraction algorithm and feature extraction algorithm are implemented in Python 2.7 and C language. In a Intel Core i7 Q720 1.6 GHz and 8GB of RAM it takes approximately 20 s to extract the PRNU and compute the features for a 1024 × 1024 crop of an image anf 5 s for a 512 × 512 crop of an image using adaptative variance estimation and zero meaning. The same case with no adaptative variance takes approximately 5 s and 1.5 s for 1024 × 1024 and 512 × 512 crops, respectively. Training the SVM classifier and testing for 600 images is realised in one minute and a fraction of a second, respectively. A random sample of 100 images was used for training and a different random sample of

IET Comput. Vis., 2015, Vol. 9, Iss. 5, pp. 723 731

Table 1 Parameters used in the proposed algorithm and its possible values

Parameter	Possible values
number of training photos by camera	100
number of testing photos by camera	100
image crop	centre: 1024 × 1024 or 512 × 512
variance estimation	adaptative (steps 7 and 8 of Fig. 4) or non adaptive (step 9 of Fig. 4)
zero meaning	ysed or not used (step 13 of Fig. 4)

Table 2 Configurations used in mobile device digital cameras

Brand	Model	Resolution	Taking Conditions
Apple	iPhone 3G (A1)	2 MP (1600 × 1200)	scene type: any
	iPhone 4S (A2)	8 MP (3264 × 2448)	orientation:
	iPhone 3 (A3)	2 MP (1600 × 1200)	vertical
	iPhone 5 (A4)	8 MP (3264 × 2448)	flash: disabled
Black Berry	8520 (B1)	2 MP (1600 × 1200)	light: natural
Sony	UST25a (SE1)	5 MP (2592 × 1944)	white balance:
Ericsson	U5I (SE2)	8 MP (3264 × 2448)	auto
Samsung	GT I9100 (S1)	8 MP (3264 × 2448)	digital zoom
	GT S5830 (S2)	5 MP (2592 × 1944)	ratio: 0
	GT S5830M (S3)	5 MP (2592 × 1944)	exposure time: 0
	EK GC101 (S4)	16.3 MP (4608 × 3456)	seg
LG	E400 (L1)	3.2 MP (2048 × 1536)	ISO speed:
	P760 (L2)	5 MP (2592 × 1944)	automatic
HTC	Desire HD (H1)	8 MP (3264 × 2448)	
	Desire (H2)	5 MP (2592 × 1944)	
Nokia	E61I (N1)	2 MP (1600 × 1200)	
	800 Lumia (N2)	8 MP (3264 × 2448)	
Zopo	ZP979 (Z1)	12.6 MP (4096 × 3072)	

100 images was used for testing. However, we used EOLO the HPC of Climate Change of the International Campus of Excellence of Moncloa for computing.

The first experiment of [28] shows that the performance changed only slightly in different experiment runs, which indicates stability over different training and testing image sets.

In experiments 1 to 8 we used the same number of phones and the same brands and models. This allows us to perform a comparative study and to obtain conclusions about what parameters can be favourable or optimal in different situations.

All the mobile devices used are shown in Table 2.

Once they have been presented with configuration parameters and cameras, the experiments with their corresponding parameters are shown in Table 3.

5.1 Experiment 1

The parameters chosen for this experiment are: crop centre 1024 × 1024, variance estimation adaptative and zero meaning.

The confusion table from six cameras is showed in Table 4. The average accuracy rate for correctly identifying camera make and model for this experiment was 96.33%.

5.2 Experiment 2

The parameters chosen for this experiment are: crop centre 1024 × 1024, variance estimation adaptative and no zero meaning. That is, the same parameters as in Experiment 1, except that in this experiment the zero meaning does not apply. Given that the images used for all experiments are the same we will be able to check the impact of this change to the results.

The confusion table from six cameras is showed in Table 5. The average accuracy rate for correctly identifying camera make and model for this experiment was 98%.

It is noted that the zero meaning gets worse the average accuracy rate (1.67% from Experiment 1), although the difference is not very significant to obtain definitive conclusions. It can also be noted that except for the model LG P760 (passing from 100 to 99%) the rest of the mobile devices increases the hit rate.

5.3 Experiment 3

The parameters chosen for this experiment are: crop centre 1024 × 1024, variance estimation non adaptative and zero meaning. That is, the same parameters as in Experiment 1, except that in this experiment the variance estimation adaptative does not apply. Among others, the main objective of this experiment is to check if the chosen type of variance estimation is determinant in the results of the algorithm. It is also important to note that the use of adaptive or non adaptive variance has important effects on the execution time of the algorithm, because algorithm execution time with non adaptative variance is approximately four times faster.

The confusion table from six cameras is showed in Table 6. The average accuracy rate for correctly identifying camera make and model for this experiment was 97.5%. At first it was expected that non adaptive variance estimation would produce worse results, but

Table 3 Parameter configuration of experiments

Experiment	Resolution	Number of devices	Multiple neighbour	Zero mean required	Average accuracy
test 1	1024 × 1024	6	t	t	96.33
test 2	1024 × 1024	6	t	f	98
test 3	1024 × 1024	6	f	t	97.5
test 4	1024 × 1024	6	f	f	97.83
test 5	512 × 512	6	t	t	73.76
test 6	512 × 512	6	t	f	93.17
test 7	512 × 512	6	f	t	92.5
test 8	512 × 512	6	f	f	91.67
test 9	1024 × 1024	14	f	f	87.21

Table 4 Experiment 1

Camera	Apple iPhone 5	Samsung EK GC101	Nokia 800 Lumia	Zopo ZP979	LG P760	Sony Ericsson ST25A
Apple iPhone 5	96	0	2	0	2	0
Samsung EK GC101	5	88	2	2	3	0
Nokia 800 Lumia	0	0	100	0	0	0
Zopo ZP979	0	0	2	98	0	0
LG P760	0	0	0	0	100	0
Sony Ericsson ST25A	0	0	3	0	1	96

Table 5 Experiment 2

Camera	Apple iPhone 5	Samsung EK GC101	Nokia 800 Lumia	Zopo ZP979	LG P760	Sony Ericsson ST25A
Apple iPhone 5	97	0	1	0	2	0
Samsung EK GC101	1	95	0	3	1	0
Nokia 800 Lumia	0	0	100	0	0	0
Zopo ZP979	0	0	2	98	0	0
LG P760	0	0	0	0	99	1
Sony Ericsson ST25A	0	0	0	0	1	99

Table 6 Experiment 3

Camera	Apple iPhone 5	Samsung EK GC101	Nokia 800 Lumia	Zopo ZP979	LG P760	Sony Ericsson ST25A
Apple iPhone 5	95	0	2	0	3	0
Samsung EK GC101	1	95	0	3	1	0
Nokia 800 Lumia	0	0	100	0	0	0
Zopo ZP979	0	1	1	98	0	0
LG P760	0	1	0	0	99	1
Sony Ericsson ST25A	0	0	1	0	1	98

it is observed that the results of the above experiments do not differ by far.

5.4 Experiment 4

The parameters chosen for this experiment are: crop centre 1024 × 1024, non adaptive variance estimation and no zero meaning. That is, the same parameters as in Experiment 2, except that in this experiment we apply the non adaptive variance estimation. Similar to the previous experiment one of the objectives of this experiment is to check if the chosen type of variance estimation has effects in the results. Besides we can watch the behaviour of zero meaning for non adaptive variance estimation.

The confusion table from six cameras is showed in Table 7. The average accuracy rate for correctly identifying camera make and model for this experiment was 9783%.

In contrast to what occurs between Experiments 1 and 3, in this experiment a small worsening on the average accuracy rate of Experiment 2 is observed. Therefore, it can be concluded that in the case of 1024 × 1024 crop using adaptive variance estimation does not improves significantly the results, because the results are almost the same with minor improvements or deteriorations. Moreover it is observed that the use of zero meaning with non adaptive variance estimation does not significantly improve the results.

5.5 Experiment 5

The parameters chosen for this experiment are: crop centre 512 × 512, variance estimation adaptative and zero meaning. That is, the same parameters as in Experiment 1, except that in this experiment the crop size is reduced. One of the aims of this experiment and the following three is to check the influence of the crop sizes in the results with different parameters.

The confusion table from six cameras is shown in Table 8. The average accuracy rate for correctly identifying camera make and model for this experiment was 89.33%. As expected, the average accuracy rate is down considerably (by 7%) relative to Experiment 1, because the amount of information used to obtain the image features is considerably less.

5.6 Experiment 6

The parameters chosen for this experiment are: crop centre 512 × 512, variance estimation adaptative and no zero meaning. That is, the same parameters as in Experiment 5, except that in this experiment zero meaning does not apply. This experiment has among others aims seeing the influence of zero meaning in small crops using adaptive variance estimation.

The confusion table from six cameras is shown in Table 9. The average accuracy rate for correctly identifying camera make and model for this experiment was 93.17%. As expected, the average

Table 7 Experiment 4

Camera	Apple iPhone 5	Samsung EK GC101	Nokia 800 Lumia	Zopo ZP979	LG P760	Sony EricssonST25A
Apple iPhone 5	96	2	0	0	2	0
Samsung EK GC101	1	95	0	3	1	0
Nokia 800 Lumia	0	0	100	0	0	0
Zopo ZP979	0	0	2	98	0	0
LG P760	0	2	0	0	99	0
Sony Ericsson ST25A	0	0	1	0	0	100

Table 8 Experiment 5

Camera	Apple iPhone 5	Samsung EK GC101	Nokia 800 Lumia	Zopo ZP979	LG P760	Sony Ericsson ST25A
Apple iPhone 5	93	3	2	0	2	0
Samsung EK GC101	16	76	2	0	6	0
Nokia 800 Lumia	0	0	86	0	2	12
Zopo ZP979	0	19	2	0	79	0
LG P760	2	0	1	0	93	4
Sony Ericsson ST25A	0	0	4	0	2	94

IET Comput. Vis., 2015, Vol. 9, Iss. 5, pp. 723 731

Table 9 Experiment 6

Camera	Apple iPhone 5	Samsung EK GC101	Nokia 800 Lumia	Zopo ZP979	LG P760	Sony Ericsson ST25A
Apple iPhone 5	94	2	3	0	1	0
Samsung EK GC101	6	91	1	1	1	0
Nokia 800 Lumia	0	0	93	0	0	7
Zopo ZP979	0	5	2	92	1	0
LG P760	2	0	0	0	95	3
Sony Ericsson ST25A	0	0	4	0	2	94

Table 10 Experiment 7

Camera	Apple iPhone 5	Samsung EK GC101	Nokia 800 Lumia	Zopo ZP979	LG P760	Sony Ericsson ST25A
Apple iPhone 5	95	0	2	0	3	0
Samsung EK GC101	5	89	0	3	3	0
Nokia 800 Lumia	0	0	85	0	1	14
Zopo ZP979	0	1	2	97	0	0
LG P760	2	0	2	0	93	3
Sony Ericsson ST25A	0	0	4	0	0	96

Table 11 Experiment 8

Camera	Apple iPhone 5	Samsung EK GC101	Nokia 800 Lumia	Zopo ZP979	LG P760	Sony Ericsson ST25A
Apple iPhone 5	95	0	2	0	3	0
Samsung EK GC101	5	89	0	3	3	0
Nokia 800 Lumia	0	0	85	0	1	14
Zopo ZP979	0	1	2	97	0	0
LG P760	2	0	2	0	93	3
Sony Ericsson ST25A	0	0	4	0	0	96

accuracy rate is down (4.83%) relative to Experiment 2, because of the reduction of crop size. In the case of smaller crop not using zero meaning, the success rate increases compared with the previous experiment (3.84%), although this increase is not a significant improvement.

5.7 Experiment 7

The parameters chosen for this experiment are: crop centre 512 × 512, non adaptive variance estimation and zero meaning. That is, the same parameters as in Experiment 5, except that in this experiment we apply non adaptive variance estimation. One of the aims of this experiment is to see the influence of the adaptive variance estimation using small crops.

The confusion table from six cameras is shown in Table 10. The average accuracy rate for correctly identifying camera make and model for this experiment was 92.50%. As expected, the average accuracy rate is down (5%) relative to Experiment 3, because of the reduction of crop size. Relative to the comparison with experiment 5, it can be seen that the results are better with non adaptive variance (3.17%). Moreover, relative to the comparison with Experiment 6 which also uses adaptive variance estimation the impact in results is minimal.

5.8 Experiment 8

The parameters chosen for this experiment are: crop centre 512 × 512, variance estimation non adaptive and no zero meaning. That is, the same parameters as in Experiment 4, except that in this experiment the crop size is reduced. One of the aims of this experiment is to see the influence of zero meaning using small crops and adaptive variance estimation.

The confusion table from six cameras is shown in Table 11. The average accuracy rate for correctly identifying camera make and model for this experiment was 91.67%. As expected, the average accuracy rate is down (6.16%) relative to Experiment 4, because of the reduction of crop size. It is confirmed that the results obtained in this experiment and the results obtained between the comparison of the results of Experiments 6 and 8 show that the use of adaptive variance estimation does not significantly improve the results.

Table 12 Confusion matrix of Experiment 9

Camera	A1	A2	A3	A4	B1	SE1	SE1	S1	S1	S3	L1	H1	H2	N1
A1	90	0	0	2	0	0	0	0	7	0	1	0	0	0
A2	0	91	0	3	0	0	0	3	0	0	0	1	2	0
A3	0	0	98	0	0	0	0	2	0	0	0	0	0	0
A4	0	0	1	88	0	0	0	0	0	0	3	6	0	2
B1	0	0	0	2	73	0	0	0	4	0	0	1	0	20
SE1	7	0	0	0	0	80	0	0	0	0	1	12	0	0
SE2	1	0	0	2	2	0	86	1	2	5	1	0	0	0
S1	4	5	0	4	0	0	1	83	0	0	1	0	2	0
S2	0	0	0	0	0	0	0	0	100	0	0	0	0	0
S3	0	0	1	0	0	0	8	0	0	85	0	1	0	5
L1	0	0	0	9	0	6	0	0	2	0	70	13	0	0
H1	2	0	0	0	0	11	0	0	1	0	1	85	0	0
H2	0	6	0	0	0	0	0	0	0	0	0	0	94	0
N1	0	0	0	0	2	0	0	0	0	0	0	0	0	98

5.9 Experiment 9

To evaluate the scalability of the method to a larger number of classes, a group of 14 mobile device digital cameras from seven different manufacturers was used. The average classification rate dropped to 87.21% as shown in the confusion matrix of Table 12 indicating a small loss in performance when the number of classes (cameras) is increased.

Remember that in all of this work, 100 images were employed for training and 100 for testing.

6 Conclusions

According to the structure and operation of mobile device digital cameras, the most appropriate techniques for forensic analysis are those based on sensor noise and wavelet transforms. In this paper, an algorithm was proposed for identifying the mobile source combining techniques based on sensor fingerprint and the wavelet transforms. The algorithm is mainly composed of two phases: the first is dedicated to extracting the sensor fingerprint, and the second to extracting features from this fingerprint that will serve as input to the SVM used as classification method.

A method for source camera identification, based on wavelet features of image noise residuals and SVM classification, was tested on photographs acquired from a range of smartphones. Eight experiments have been made with the same pictures, for the purpose of analysing the different configuration parameters and improvements in the used algorithm, which allow it to adapt to different situations. First, in general, note that the best results obtained have an average accuracy rate of 98% and the worst of 89.33%. This wide range implies that the possibility exists to set parameters to improve the algorithm for each situation.

Then, the general conclusions are presented after the previous analysis of the experiments.

The first expected conclusion is that regardless of the parameters used in the algorithm, we obtain worse results as the used crop is smaller. There is not a case in the experiments that the average accuracy rate with a small crop exceeds the worst results with a big crop for the same number of devices. Obviously, the processing in terms of execution time increases as higher crop is used.

The second general conclusion is that there are not clearly defined configuration parameters for the algorithm for each crop size that allows the best results to be obtained. Any obtained combination of parameters has similar results, although it is noteworthy that there are parameters that optimise the average accuracy rate to a greater extent. It is the responsibility of the forensic analysts to achieve greater results optimisation at the expense of a longer execution time or otherwise. Moreover, it can be concluded that none of the parameters used are superfluous because none of them independently weaken the results for all possible combinations.

A third general conclusion is that for both large and small crops there is a common configuration that gets the best results: adaptive variance estimation and no zero meaning.

Focusing on the case of each crop size, the conclusions are shown below.

For the case of large crops (1024 × 1024) it can be concluded that the use of different configuration parameters does not clearly generate better results compared with the other options (the largest difference between all the results is 1.67%). The best option is to use adaptive variance estimation and not zero meaning and the second best option does not use zero meaning either. Hence, we can conclude that for large crops the zero meaning does not provide any improvement and it makes the results slightly worse. Regarding the type of variance to use, it can be concluded that taking into account the processing time using adaptive variance it takes a long time. For large crops and a large number of images to be analysed it is better not to use it (in the worst case the results worsen by 0.5%), unless there are not time restrictions or we have high throughput.

In the case of small crops (512 × 512), there are no significant differences with respect to the use of different configuration parameters. The worst case is the one that uses the adaptive variance estimation and zero meaning; in small crops we conclude that it is a bad choice because it gets far worse results than the other options (2.34% in the best case).

Concerning the use of various types of variance estimation and zero meaning conclusions are similar to the case of large crops.

To evaluate the scalability of the approach, we repeated the experiment using 14 models from seven manufactures and achieved an average success rate of 87.21%.

Depending on the number and the type of images that have to be analysed and maximising the success rate depending on the desire processing time, the forensic analyst has the possibility of setting certain parameters in the algorithm of identifying the source acquisition. This will allow the analyst to obtain results closer to their needs and processing constraints.

Our results, tentatively, suggest that the method is applicable to datasets containing images from a large number of different cameras and therefore the method promises potential uses for digital forensics and data mining applications.

7 Acknowledgments

Part of the computations of this work were performed in EOLO, the HPC of Climate Change of the International Campus of Excellence of Moncloa, funded by MECD and MICINN. This is a contribution to CEI Moncloa.

8 References

1 'Gartner says smartphone sales grew 46.5 percent in second quarter of 2013 and exceeded feature phone sales for first time', http://www.gartner.com/newsroom/id/2573415, 2013
2 Ahonen, T., Moore, A., Almanac, T.A.: 'Mobile telecoms industry annual review', 2012
3 Embedded imaging takes off as stand-alone digital cameras stall, http://www.icinsights.com/news/bulletins/Embedded-Imaging-Takes-Off-As-Standalone-Digital-Cameras-Stall/, 2013
4 Baer, R.: 'Resolution limits in digital photography: the looming end of the pixel wars – OSA technical digest (CD)'. Proc. Imaging Systems, Optical Society of America, June 2010, p. ITuB3
5 Al-Zarouni, M.: 'Mobile handset forensic evidence: a challenge for law enforcement'. Proc. Fourth Australian Digital Forensics Conference, School of Computer and Information Science, Edith Cowan University, December 2006
6 Gloe, T., Kirchner, M., Winkler, A., Bohme, R.: 'Can we trust digital image forensics?'. Proc. 15th Int. Conf. on Multimedia, September 2007, pp. 78–86
7 Ramanath, R., Snyder, W.E., Yoo, Y., Drew, M.S.: 'Color image processing pipeline', *IEEE Signal Process. Mag.*, 2005, **22**, (1), pp. 34–43
8 Mckay, C., Swaminathan, A., Gou, H., Wu, M.: 'Image acquisition forensics: forensic analysis to identify imaging source'. Proc. IEEE Int. Conf. on Acoustics, Speech and Signal Processing, June 2008, pp. 1657–1660
9 Wang, B., Guo, Y., Kong, X., Meng, F.: 'Source camera identification forensics based on wavelet features'. Proc. Int. Conf. on Intelligent Information Hiding and Multimedia Signal Processing, IEEE Computer Society, September 2009, pp. 702–705
10 Van Lanh, T., Chong, K.S., Emmanuel, S., Kankanhalli, M.S.: 'A survey on digital camera image forensic methods'. Proc. IEEE Int. Conf. on Multimedia and Expo, July 2007, pp. 16–19
11 Choi, K.S.: 'Source camera identification using footprints from lens aberration'. Proc. Digital Photography II, number 852 in 6069, SPIE Int. Society For Optical Engineering, February 2006, pp. 60690J–60690J-8
12 Bayram, S., Sencar, H.T., Memon, N.: 'Classification of digital camera-models based on demosaicing artifacts', *Digit. Invest.*, 2008, **5**, (1–2), pp. 49–59
13 Long, Y., Huang, Y.: 'Image based source camera identification using demosaicking'. Proc. IEEE Eighth Workshop on Multimedia Signal Processing, October 2006, pp. 419–424
14 Celiktutan, O., Avcibas, I., Sankur, B., Ayerden, N.P., Capar, C.: 'Source cell-phone identification'. Proc. IEEE 14th Signal Processing and Communications Applications, April 2006, pp. 1–3
15 Meng, F.J., Kong, X.W., You, X.G.: 'Source camera identification based on image bi-coherence and wavelet features'. Proc. Fourth Annual IFIP WG 11.9 Int. Conf. on Digital Forensics, Kyoto, Japan, January 2008
16 Ozparlak, L., Avcibas, I.: 'Differentiating between images using wavelet-based transforms: A comparative study', *IEEE Trans. Inf. Forensics Sec.*, 2011, **6**, (4), pp. 1418–1431
17 Liu, Q., Li, X., Chen, L., *et al.*: 'Identification of smartphone-image source and manipulation', in Jiang, H., Ding, W., Ali, M., Wu, X. (Eds.): 'Advanced

research in applied artificial intelligence' (Springer, Berlin Heidelberg, Dalian, China, 2012), pp. 262–271
18 Geradts, Z.J., Bijhold, J., Kieft, M., Kurosawa, K., Kuroki, K., Saitoh, N.: 'Methods for identification of images acquired with digital cameras'. Proc. Enabling Technologies for Law Enforcement and Security, SPIE-Int. Society for Optical Engine, February 2001, vol. 4232, pp. 505–512
19 Lukas, J., Fridrich, J., Goljan, M.: 'Digital camera identification from sensor pattern noise', *IEEE Trans. Inf. Forensics Sec.*, 2006, **1**, (2), pp. 205–214
20 Costa, F.D.O., Eckmann, M., Scheirer, W.J., Rocha, A.: 'Open set source camera attribution'. Proc. 25th Conf. on Graphics, Patterns and Images, IEEE, August 2012, pp. 71–78
21 Li, C.T.: 'Source camera linking using enhanced sensor pattern noise extracted from images'. Proc. Third Int. Conf. on Crime Detection and Prevention (ICDP 2009), Curran Associates, Inc., December 2009, pp. 1–6
22 Sandoval Orozco, A.L., Arenas Gonźalez, D.M., Corripio, J.R., Garćia Villalba, L. J., Hernandez-Castro, J.C.: 'Techniques for source camera identification'. Proc. Sixth Int. Conf. on Information Technology, May 2013, pp. 1–9
23 Chen, M., Fridrich, J., Goljan, M., Lukas, J.: 'Determining image origin and integrity using sensor noise', *IEEE Trans. Inf. Forensics Sec.*, 2008, **3**, (1), pp. 74–90
24 Celiktutan, O., Sankur, B., Avcibas, I.: 'Blind identification of source cell-phone model', *IEEE Trans. Inf. Forensics Sec.*, 2008, **3**, (3), pp. 553–566
25 McKay, C.: 'Forensic analysis of digital imaging devices'. Technical report, University of Maryland, 2007
26 Adams, J., Parulski, K., Spaulding, K.: 'Color processing in digital cameras', *IEEE Micro,* 1998, **18**, (6), pp. 20–30
27 Chang, C.C., Lin, C.J.: 'LIBSVM: A library for support vector machines', Version 3.17, 26 April 2013, http://www.csie.ntu.edu.tw/cjlin/libsvm/
28 Corripio, J.R., Arenas Gonźalez, D.M., Sandoval Orozco, A.L., Garćia Villalba, L. J., Hernandez-Castro, J.C., Gibson, S.J.: 'Source smartphone identification using sensor pattern noise and wavelet transform'. Proc. Fifth Int. Conf. on Imaging for Crime Detection and Prevention (ICDP 2013), 16–17 December 2013, pp. 1–6

MEMORIAS

VIII CONGRESO IBEROAMERICANO DE SEGURIDAD INFORMÁTICA

III Taller Iberoamericano de enseñanza e innovación educativa en seguridad de la información

CIBSI
ECUADOR 2015

10-12 NOV 2015
UNIVERSIDAD DE LAS FUERZAS ARMADAS DEL ECUADOR - ESPE
Sangolquí, ECUADOR

ESPE
UNIVERSIDAD DE LAS FUERZAS ARMADAS
INNOVACIÓN PARA LA EXCELENCIA

Con la Organización de

ESPE-Innovativa EMPRESA PÚBLICA
CRIPTORED
fundación in-n-va Centro de innovación

Memorias del VIII Congreso Iberoamericano de Seguridad Informática

CIBSI 2015

Sangolqui (Quito), Ecuador, 10 al 12 de Noviembre del 2015

Compiladores

Luis Enrique Sánchez Crespo

Walter Marcelo Fuertes Díaz

Jorge Ramió Aguirre

ISBN: 978-9978-301-61-6

9 789978 301616

Universidad De Las Fuerzas Armadas Del Ecuador -ESPE

Quito, Ecuador

Método Anti-Forense para Manipular la Fuente de Adquisición de una Imagen de Dispositivo Móvil

Jocelin Rosales Corripio, Anissa El-Khattabi, Ana Lucila Sandoval Orozco, Luis Javier García Villalba

***Abstract*—Nowadays digital images play an important role in our society. The mobile device camera presence is growing at an unstoppable rate, causing that most of digital images come from this kind of devices. While the developing technology makes image generation process easier, at the same time it facilitates forgery; therefore, image forensics is gaining relevance. In this paper we propose a pair of algorithms that are based on sensor noise and the wavelet transform, the first to eliminate the possibility of identifying the mobile device (maker and model) that generated an image and the second to forge the identity of a given image.**

***Index Terms*—Counter Forensics, Forensics Analysis, Image Anonymity, Image Forgery, PRNU, Wavelet.**

I. Introducción

Aunque las imágenes pueden ser consideradas parte de la verdad, ya que son hechos reales captados por dispositivos electrónicos (cámaras), nunca ha sido tan fácil modificar las imágenes como lo es hoy en día, dada la existencia de potentes y sofisticados programas software. Esta facilidad de manipulación plantea interrogantes sobre la integridad y veracidad de las imágenes.

Actualmente, las ventas de dispositivos móviles (teléfonos, *smartphones*, PDAs, *tablets*, etc.) siguen aumentando incluso con el impacto de la crisis financiera global. La inmensa mayoría, concretamente el 83 % de los teléfonos móviles en 2012, tienen una cámara fotográfica integrada. Las cámaras integradas en dispositivos móviles ya superan en número a las cámaras de fotos tradicionales o *Digital Still Camera* (DSCs). En total, según estimaciones de la Unión Internacional de Telecomunicaciones (UIT), hay 6,8 miles de millones de suscripciones de teléfonos móviles en todo el mundo, lo cual supone un gran incremento sobre los 6000 millones de suscripciones de 2012 y 5800 millones de 2011. De igual modo existen predicciones para el futuro que indican que las DSCs desaparecerán en pro de las nuevas integradas en dispositivos móviles, ya que el aumento de calidad de estas cámaras crece a un ritmo imparable.

En nuestro día a día es habitual ver cómo se realizan y usan fotografías de este tipo de dispositivos para una gran diversidad de situaciones (vida personal, noticias, pruebas judiciales, aplicaciones para teléfonos móviles, etc.).

Jocelin Rosales Corripio, Anissa El-Khattabi, Ana Lucila Sandoval Orozco y Luis Javier García Villalba, Grupo de Análisis, Seguridad y Sistemas (GASS, http://gass.ucm.es), Departamento de Ingeniería del Software e Inteligencia Artificial (DISIA), Facultad de Informática, Despacho 431, Universidad Complutense de Madrid (UCM), Calle Profesor José García Santesmases, 9, Ciudad Universitaria, 28040 Madrid, España. E-mail: jocelinr@ucm.es, {asandoval, javiergv}@fdi.ucm.es.

Este progreso hace que el tratamiento y la toma de fotografías con este tipo de dispositivos puedan crear situaciones problemáticas o beneficiosas en las distintas realidades. Muchos estiman que este tipo de cámaras facilitan la proliferación de crímenes contra la privacidad y la seguridad de la información (robo con tarjetas de crédito, pornografía infantil, espionaje industrial, etc.). De hecho, una de las principales razones de la existencia hoy en día de dispositivos sin cámaras fotográficas se debe a que diversas compañías, organizaciones o países poseen normas que prohíben o limitan su uso [1].

Una consecuencia más de su extenso uso es que las imágenes digitales en la actualidad son utilizadas como testigos silenciosos en procesos judiciales, siendo una pieza crucial de la evidencia del crimen [1]. Debido a esto muchas áreas pueden beneficiarse del análisis forense de imágenes, tales como la lucha contra la pornografía infantil, la prevención de robo de tarjetas de crédito, el combate a la piratería, la prevención de secuestros, etc.

Por todo lo anterior, el análisis forense de imágenes digitales se ha convertido en un tema de interés en los últimos años. El análisis forense surge con la idea de restablecer la confiabilidad en las imágenes digitales que de otro modo se consideraban muy fácilmente modificables. En sus inicios, la parte académica encontró el análisis forense útil en áreas como el uso de imágenes para aplicaciones legales, inteligencia, investigaciones privadas y medios de comunicación.

Como en la mayoría de campos de estudio existe una contracorriente, en este caso, personas como espías o estafadores que hacen esfuerzos para manipular las imágenes en su propio beneficio usando el conocimiento del análisis forense de imágenes para borrar o incluso suplantar las huellas o rastros que se utilizan para determinar la identidad de las imágenes. Muchos de los algoritmos forenses existentes en la literatura no fueron diseñados teniendo en cuenta ese tipo de comportamiento y como consecuencia son fáciles de "engañar".

La posibilidad de copiar las huellas digitales de una imagen se puede convertir en un ciclo infinito que puede permitir que personas inocentes sean inculpadas o que criminales aseguren que las pruebas son resultados de una falsificación. Al final, la confianza en las técnicas forenses de imágenes se podría ver comprometida. Es por esto que surge la necesidad de considerar los posibles ataques en el momento de diseñar técnicas de análisis forense en imágenes digitales.

Así como en el área de seguridad el estudio de los ataques permite mejorarla, los métodos forenses de imágenes se pueden beneficiar del estudio de las técnicas de ataque para robustecer los algoritmos de las próximas generaciones.

Este documento se estructura en 7 secciones, siendo la primera la presente introducción. La sección II explica brevemente el proceso de formación de una imagen digital. La sección III resume los principales trabajos relacionados con el análisis forense de imágenes de dispositivos móviles. Los ataques a las técnicas de análisis forense de imágenes digitales se muestran en la sección IV. En la sección V se especifica el algoritmo de falsificación de la identidad de una imagen propuesto. La sección VI describe la experimentación realizada y se analizan los resultados obtenidos en la misma. Finalmente, la sección VII contiene las principales conclusiones del trabajo.

II. Formación de una Imagen Digital

Para la comprensión del análisis forense de imágenes digitales es fundamental conocer en detalle el proceso de adquisición de imágenes en las cámaras digitales. Este puede resumirse en la Figura 1.

Aunque muchos de los detalles del *pipeline* de una cámara pertenecen a cada fabricante, la estructura general es la misma en todas ellas. El *pipeline* de una cámara digital consiste básicamente en un sistema de lentes, un conjunto de filtros, una matriz de filtros de color o *Color Filter Array* (CFA), un sensor de imagen y un procesador de imagen digital o *Digital Image Processor* (DIP).

Para generar una imagen digital, en primer lugar, el sistema de lentes recoge la luz de la escena controlando la exposición, el enfoque y la estabilización de imagen. Seguidamente, la luz entra en la cámara a través de la lente, pasando por una combinación de filtros (por lo menos infrarrojos y *anti-aliasing*) para garantizar la máxima calidad de la imagen. Con el objetivo de producir una imagen en color se utiliza la CFA. Después, la luz se enfoca sobre el sensor de imagen que es una matriz de elementos sensibles a la luz llamados píxeles. La incidencia de la luz contra los píxeles genera una señal analógica proporcional a la intensidad de la luz, la cual se convierte en una señal digital para ser procesada por el DIP. Finalmente, la imagen final completa se forma por el DIP, el cual lleva a cabo algunas operaciones tales como *demosaicing*, corrección de puntos blancos, corrección *gamma*, compresión, etc., con el objetivo de producir una imagen visualmente agradable.

III. Técnicas de Análisis Forense en Imágenes

Las tareas de análisis forense de imágenes digitales se dividen, de acuerdo a su objetivo, en las siguientes ramas [2]: verificación de integridad, recuperación de la historia de procesamiento, clasificación basada en la fuente, agrupación por dispositivo fuente e identificación de la fuente [3].

Para el diseño de técnicas y algoritmos en cualquiera de estas ramas se aprovechan algunas características especiales de las imágenes creadas con móviles que sirven como herramienta para el análisis forense. [4] y [5] realizan un estudio de las características que pueden ser objeto de análisis forense en dispositivos móviles.

Con respecto a la rama de identificación de la fuente los estudios realizados hasta el momento en esta área se dividen básicamente en cuatro grupos dependiendo de la información que se utiliza como base para identificar la fuente [4]: los basados en las aberraciones del sistema de lentes, los basados en la elección de la CFA y en la especificación del algoritmo de interpolación de color [6], los basados en características de la imagen (color, métricas de calidad y dominio de la frecuencia) [7] [8] y los basados en el ruido del sensor *Photo Response Non Uniformity* (PRNU) [9] [10].

Este documento se centra en los ataques contra una de las técnicas en el campo de la identificación de la fuente: la basada en el ruido del sensor. Estas técnicas se dividen principalmente en dos ramas: defectos de píxel y patrón de ruido del sensor o *Sensor Pattern Noise* (SPN). En la primera se estudian los defectos de píxel, los píxeles calientes, los píxeles muertos, los defectos de fila o columna, y los defectos de grupo. En la segunda se construye un patrón del ruido promediando los múltiples residuos de ruido obtenidos mediante algún filtro de eliminación de ruido. La presencia del patrón se determina utilizando algún método de clasificación como correlación o máquinas *Support Vector Machine* (SVM).

Geradts et al. [11] estudian los defectos de los píxeles en los sensores de tipo *Charge Coupled Device* (CCD), centrándose en la evaluación de diferentes características para examinar las imágenes e identificar la fuente: defectos del sensor CCD, formato de los archivos usados, ruido introducido en la imagen y marcas de agua introducidas por el fabricante de la cámara. Entre los defectos del sensor CCD considerados se encuentran los puntos calientes, los píxeles muertos, los defectos en grupo y los defectos de fila o columna. En sus resultados se observa que cada una de las cámaras tiene un patrón de defecto diferente. Sin embargo, también se señala que el número de defectos en los píxeles para una cámara es diferente entre fotos y varía demasiado en función del contenido de la imagen. Asimismo, se revela que el número de defectos cambia con la temperatura. Por último, el estudio encontró que las cámaras con CCD de alta calidad no tienen este tipo de problema.

También es cierto que la mayoría de las cámaras tienen

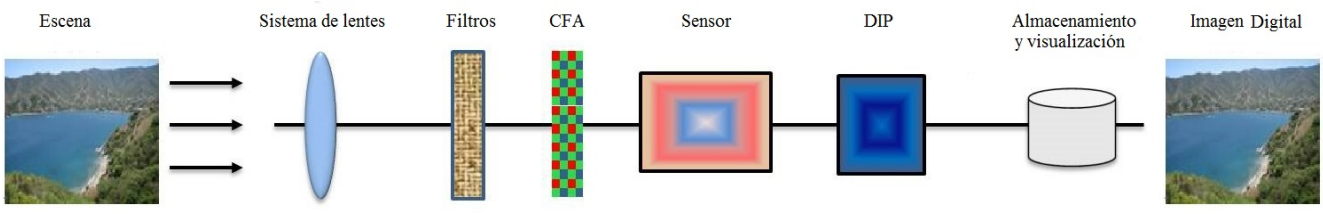


Figura 1. Proceso de generación de una imagen en una cámara digital.

mecanismos adicionales para compensar este tipo de problemas. Al considerar únicamente los defectos de los sensores de tipo CCD, este estudio no es aplicable al análisis de imágenes generadas por dispositivos móviles.

Lukas et al. [12] analizan el patrón de ruido del sensor de un conjunto de cámaras, el cual funciona como una huella dactilar, permitiendo la identificación única de cada cámara. Para obtener este patrón se realiza un promedio del ruido obtenido a partir de diferentes imágenes utilizando un filtro de eliminación de ruido. Para identificar la cámara a partir de una imagen dada, se considera el patrón de referencia como una marca de agua cuya presencia en la imagen es establecida mediante un detector de correlación. El estudio se realizó con 320 imágenes procedentes de 9 modelos distintos de cámaras. También se demuestra que este método está afectado por algoritmos de procesamiento de la imagen como la compresión *Joint Photographic Experts Group* (JPEG) y la corrección *gamma*. Los resultados para fotografías con diferentes tamaños y recortadas no son satisfactorios [4].

Costa et al. [13] proponen un enfoque para la identificación de la cámara fuente considerando escenarios abiertos donde, a diferencia de los escenarios cerrados, no se da por sentado contar con acceso a todas las posibles cámaras de origen de la imagen. Esta propuesta comprende tres fases: definición de las regiones de interés, determinación de las características e identificación de la cámara fuente. Las diferentes regiones de las imágenes pueden contener información distinta sobre la huella digital de la cámara fuente. Este enfoque, en contraste con otros, considera diferentes áreas de interés o *Region Of Interest* (ROI) y no sólo la región central de la imagen. Para cada imagen se definen nueve ROIs. Se asume que estas regiones coinciden con el eje principal de la lente y, por lo tanto, deben tener más detalles de la escena porque los fotógrafos aficionados por lo general centran el objeto de interés en el centro de la lente. Un aspecto importante a tener en cuenta es que el uso de las regiones de interés permite trabajar con imágenes de diferentes resoluciones sin la necesidad de rellenar con ceros las imágenes y sin el uso de artefactos de interpolación de color.

IV. Ataques al Análisis Forense de Imágenes

En comparación con el destacado papel de las imágenes digitales en la sociedad multimedia de hoy en día, la investigación en el campo de la autenticidad de la imagen se encuentra todavía en una fase muy preliminar. La mayoría de las publicaciones en este campo emergente todavía carecen de discusiones rigurosas y robustas contra los falsificadores estratégicos, que prevén la existencia de técnicas forenses [14].

El área que se encarga de estudiar ataques a las técnicas de análisis forense de imágenes es conocida como *counter-forensics*. Los ataques contra los algoritmos forenses de imágenes digitales son aquellas técnicas cuyo objetivo es confundir sistemáticamente a los procedimientos de identificación de la fuente de la imagen o de detección de manipulaciones maliciosas en las imágenes. Estos ataques pueden tener uno de los siguientes objetivos: camuflaje de post-procesamientos maliciosos sobre la imagen o manipulación de la identificación de la fuente.

A. El Camuflaje de Post-Procesamientos

Estas técnicas tienen como objetivo ocultar la existencia de algún proceso aplicado a una imagen analizando los rasgos que éstos dejan sobre la imagen durante su aplicación para así poder contrarrestarlos. En [15] se estudia las dependencias introducidas durante el re-dimensionamiento o la rotación de las imágenes. En [16] se estudian los coeficientes estadísticos de los JPEG para detectar la re-compresión. En [17] se analiza la fase de congruencia para detectar la composición de imágenes a través del recortado y pegado de diferentes imágenes.

En [14] se presenta una propuesta para ocultar el proceso de re-muestreo (*resampling*). El re-muestreo es el redimensionamiento con interpolación de las imágenes. Este proceso es muy común en las operaciones primitivas de imágenes como escalamiento y rotación. Los algoritmos detectores de re-muestreo se basan en la búsqueda de las dependencias sistemáticas y periódicas entre píxeles vecinos insertadas cuando se aplica la operación de re-muestreo. Para ocultar el re-muestreo es necesario romper las equidistancias periódicas introduciendo distorsiones geométricas conocidas como ataques de marca de agua. En este caso se superpone un vector de distorsión aleatoria a las posiciones de cada píxel donde un parámetro determina el grado de distorsión introducido. Para evitar generar características visibles en la imagen como ruido se debe modular la fuerza de la distorsión empleando dos detectores de bordes: uno en dirección vertical y otro en dirección horizontal.

B. Manipulación de la Identificación de la Fuente

Así como para el proceso de identificación de la fuente se usa la extracción del ruido del sensor en la imagen, un contraataque lógico para esta técnica consta de la eliminación del ruido del sensor. Dando un paso más adelante se puede pensar también en la posibilidad de eliminar el ruido del sensor de la imagen y sustituirlo por el ruido del sensor que pertenezca a otra cámara.

1) Destrucción de la Identidad de una Imagen: En [14] se demostró que la resta de las características del dominio *wavelet* de las imágenes no es suficiente para eliminar el ruido de una imagen, además de que este procedimiento deja rastros visibles sobre la imagen. Existe otro método bastante conocido para la eliminación del ruido de una imagen llamado corrección de sensibilidad o *flatfielding*. Este método es usado típicamente en astronomía o en el proceso de escaneado de planos para mejorar la calidad de las imágenes. El *flatfielding* se realiza en base a los principales componentes del ruido de la imagen: el ruido de patrón fijo o *Fixed Pattern Noise* (FPN) y el ruido de respuesta no uniforme o *Photo Response Non Uniformity* (PRNU). El ruido FPN se calcula en términos de un marco oscuro d promediando K imágenes x_{oscura} capturadas en un ambiente completamente oscuro que se puede emular cubriendo completamente la lente de la cámara.

El ruido PRNU se calcula en términos de un marco plano (*flatfield*) f promediando L imágenes $x_{iluminada}$ de una escena iluminada homogéneamente. A las L imágenes se les

elimina el ruido FPN mediante la resta del marco oscuro d antes de promediarlas.

Como se describe en [12] [14], los atacantes pueden intentar evitar la identificación correcta de la fuente ya que existe la posibilidad de eliminar y extraer la huella de una imagen. La destrucción de la huella de una imagen x generada con una cámara específica se realiza con la ecuación 1 restando a la imagen original x el marco oscuro d y dividiendo el resultado de la resta por el marco plano f.

$$\tilde{x} = \frac{x - d}{f} \tag{1}$$

A pesar que los resultados obtenidos con esta técnica son buenos, se presentan algunos inconvenientes:

- Llevar a cabo una corrección de sensibilidades perfecta en un gran número de fotos es difícil ya que los parámetros para calcular el PRNU y el FPN deben coincidir con los de la imagen a atacar.
- En la propuesta se asume que el atacante puede tener acceso a la cámara fuente de la imagen x para generar los marcos oscuros y planos y éste no es un escenario próximo a la realidad.

Existen otras posibilidades menos robustas para destruir la identidad que en ciertos casos podrían ser efectivas ya que no necesitan contar con imágenes procedentes de la cámara origen para generar el marco oscuro y el marco plano, pero a cambio de esta facilidad la calidad de la imagen puede verse reducida y podrían introducirse algunos rasgos visuales. Por ejemplo, es posible rotar la imagen unos pocos grados, escalar la imagen, o aplicar un filtro de desenfoque gaussiano.

2) Falsificación de la Identidad de una Imagen: De igual forma que se puede eliminar el ruido en una imagen haciendo uso de la técnica de corrección de sensibilidad, se puede inyectar el ruido de la imagen de otra cámara diferente mediante la corrección de sensibilidad inversa con la ecuación 2 [14].

$$\tilde{y} = \tilde{x} \cdot f_{falsa} + d_{falsa} \tag{2}$$

donde f_{falsa} y d_{falsa} corresponden a la cámara que se pretende plagiar y $\tilde{x}$ es la imagen original sin ruido.

En [18] se propone el algoritmo 1 para falsificar la identidad de una cámara.

Algoritmo 1: Falsificación de la identidad de una cámara

1. Calcular el promedio de las huellas $F(C1)$ de la cámara $C1$ con la que se atacará;
2. Tomar una fotografía P con la segunda cámara $C2$;
3. Sumar $F(C1)$ a la fotografía P;

En el caso de que las dimensiones de *F(C1)* y *P* no coincidan, es necesario aplicar un recorte o una reconstrucción para igualar el tamaño de las imágenes.

También se propone una mejora al algoritmo de falsificación anterior para enmascarar los rasgos de la cámara C2. Esta técnica se presenta en el algoritmo 2.

Al restar *F(C2)* se trata de eliminar la correlación entre la fotografía *P* y la cámara *C2*.

Algoritmo 2: Falsificación de la identidad de una cámara para imágenes con dimensiones diferentes

1. Calcular el promedio de las huellas $F(C1)$ de la cámara $C1$ con la que se atacará;
2. Calcular el promedio de las huellas $F(C2)$ de la cámara $C2$;
3. Sumar $F(C1)$ a la fotografía P;
4. Tomar una fotografía P con la cámara $C2$;
5. Restar $F(C2)$ a P;

V. Método Anti-Forense Basado en PRNU

En este trabajo se propone un algoritmo que permite extraer y eliminar la huella del sensor de una imagen P_1 así como inyectar el patrón del sensor de una cámara C_1 a una imagen P_2 generada con una cámara C_2 sin requerir acceso a la cámara C_2.

A. Algoritmo de Eliminación de la Huella del Sensor

Entre los diferentes filtros que existen para la eliminación del ruido de las imágenes, los que usan la transformada *wavelet* dan mejor resultado debido a que el ruido residual que se obtiene con este filtro contiene la menor cantidad de rasgos de la escena. Generalmente, las áreas alrededor de los bordes son malinterpretadas cuando se utilizan únicamente filtros de eliminación de ruido menos robustos, tales como el filtro de Wiener o el filtro de mediana. Por este motivo se seleccionó el filtro de eliminación de ruido basado en la transformada *wavelet*. El algoritmo 3, basado en las ideas de [12], muestra los pasos a seguir para eliminar la huella del sensor.

Algoritmo 3: Eliminación de la huella del sensor

Input: I es la imagen víctima

1. **procedure** RemovePRNU(I)
2. Realizar una descomposición wavelet de 4 niveles de I_n;
3. **foreach** *nivel de la descomposición wavelet* **do**
4. **foreach** $c \in \{H,V,D\}$ **do**
5. Calcular la varianza local;
6. **if** *varianza adaptativa* **then**
7. Calcular 4 varianzas con ventanas de tamaños 3, 5, 7 y 9, respectivamente;
8. Seleccionar la varianza mínima;
   else
9. Calcular la varianza con una ventana de tamaño 3;
10. Calcular los componentes wavelet sin ruido aplicando el filtro de Wiener a la varianza;
11. **end procedure**

B. Algoritmo de Falsificación de la Identidad

Las técnicas de identificación de la fuente basadas en PRNU calculan la huella del sensor de la imagen con la ecuación:

$$I_{ruido} = I - I_{limpia} \quad (3)$$

donde I_{limpia} es obtenida aplicando algunos filtros de eliminación descritos en la sección V-A. En particular, en este trabajo la eliminación de ruido se realiza aplicando el algoritmo 3.

El patrón del ruido *Pixel Non-Uniformity* (PNU) se calcula mediante el promedio del ruido residual de varias imágenes con la siguiente ecuación:

$$P_{ruido} = \frac{1}{N}\sum_{i=1}^{N} I_{ruido} \quad (4)$$

Una vez que se tiene la posibilidad de eliminar el ruido del sensor y de extraer el patrón del ruido del sensor se puede plantear la falsificación de la identidad de una imagen. El algoritmo 4 muestra los pasos a seguir para falsificar la identidad de una imagen.

Algoritmo 4: Falsificación de la identidad de una imagen

Input: I es la imagen víctima
N es el número de imágenes de superficies uniformemente iluminadas de la cámara suplantadora

1. **procedure** FORGEIMG(I, N)
2. $I_{limpia} \leftarrow$ REMOVEPRNU(I);
3. $P_{ruido} \leftarrow$ EXTRACTPRNU(N);
4. Realizar una descomposición wavelet de un nivel de I_{limpia} obteniendo los componentes L_I, H_I, V_I, D_I;
5. Realizar una descomposición wavelet de un nivel de P_{ruido} obteniendo los componentes H_P, V_P, D_P;
6. Calcular los componentes wavelet falsificados mediante $c_F = c_I + c_P$ donde $c \in \{H,V,D\}$;
7. Obtener I_{falsa} aplicando la transformada wavelet inversa con L_I, H_F, V_F, D_F;
8. **end procedure**

Para tener una huella de mayor calidad y conseguir mejores resultados en la falsificación se recomienda que el número N de imágenes sea superior a 50, y que las imágenes se hayan adquirido de superficies planas sin textura iluminadas uniformemente. Como superficies planas se pueden considerar fotografías del cielo despejado o de un papel blanco.

VI. EXPERIMENTACIÓN

En esta sección se describen los experimentos realizados con los algoritmos de eliminación de la huella del sensor (algoritmo 3) y de falsificación de la huella de la cámara fuente (algoritmo 4). En estos experimentos se realizó la eliminación y la falsificación de las huellas con los algoritmos propuestos y también con la herramienta "PRNU Decompare" [19] para realizar la comparación de los resultados. "PRNU Decompare" utiliza la técnica de *flatfielding* descrita en la sección V-A que permite la eliminación y la suplantación del patrón de la huella del sensor [19]. Esta herramienta requiere como entrada una fotografía de un marco oscuro y un número N de imágenes de superficies planas iluminadas uniformemente (se recomienda un mínimo de 30 imágenes).

Los resultados obtenidos se compararon haciendo uso la herramienta "NFI PRNU Compare" [20], la cual permite comparar los patrones del ruido del sensor de varias imágenes.

A. Eliminación de la Identidad de una Imagen

Para este experimento se utilizaron las fotografías de 3 cámaras digitales de dispositivos móviles (LG E510f, LG 400, Samsung GT-I8160P). De cada uno de los dispositivos se obtuvieron 50 fotografías de imágenes planas uniformemente iluminadas, 1 fotografía totalmente oscura cubriendo totalmente la lente de la cámara y 1 fotografía seleccionada al azar de la base de datos de fotografías. Todas las fotografías fueron recortadas a un tamaño de 1024 x 1024.

Inicialmente, se generó el primer grupo de imágenes sin huella, haciendo uso del algoritmo 3. Cabe remarcar que para realizar esta eliminación no se necesitaron fotografías adicionales a la fotografía de la que se pretendía eliminar el ruido del sensor. A continuación, se generó el segundo grupo de imágenes sin huella con la herramienta "PRNU Decompare", dando como entrada a este programa las 50 imágenes planas y la imagen del marco oscuro.

Para evaluar la efectividad del algoritmo de eliminación de la huella se compararon los dos grupos de imágenes con la herramienta "NFI PRNU Compare".

En la Tabla I se muestran los resultados de comparar cada patrón del ruido del sensor de cada una de las cámaras con las imágenes sin ruido generadas por las dos herramientas y contra la fotografía original.

Tabla I
COMPARATIVA ENTRE PATRONES E IMÁGENES SIN RUIDO

Patrón	Foto	Rojo	Verde	Azul	Suma
LG E510f	Original	-0,014645672	-0,0017777978	-0,007864626	-0,024288096
	Sin ruido - Propuesta	-0,015506644	-0,003044259	-0,008411303	-0,026962206
	Sin ruido - Decompare	-0,018929206	-0,0023383496	-0,012027217	-0,033294775
Samsung GTI8160P	Sin ruido - Propuesta	0,0051651257	0,005551344	0,0042196396	0,01493611
	Original	0,004623602	0,0050348267	0,0030041975	0,012662627
	Sin ruido - Decompare	-0,0012833464	-0,001952231	-0,0026684676	-0,005904045
LG E400	Original	0,011481647	0,010190065	0,01825918	0,039930895
	Sin ruido - Propuesta	0,010315191	0,008225861	0,017940063	0,036481116
	Sin ruido - Decompare	0,010638827	0,009045472	0,016430777	0,036115076

La herramienta “NFI PRNU Compare” nos permite hacer comparaciones midiendo qué tanto se parece un patrón a otro: las filas que están más cerca del patrón con el que se compara son las que más se le asemejan.

En las 3 pruebas las imágenes sin ruidos generadas con “PRNU Decompare” resultaron ser las menos parecidas al patrón. Esto era de esperar ya que consideran mayor número de información para eliminar la huella. De manera sorprendente para la cámara Samsung-GTI8160P la fotografía sin ruido generada por la propuesta de este trabajo resultó ser más parecida al patrón que la misma fotografía original. Posiblemente en el caso de esta fotografía el contenido de la fotografía tenga alguna influencia.

En el caso de la cámara LG-400, los resultados de las comparaciones de las imágenes sin ruido tuvieron resultados muy similares, lo que indica que en este caso el algoritmo propuesto obtuvo buenos resultados acercándose al resultado del programa “PRNU Decompare” pero sin la necesidad de usar la fotografía del marco oscuro, ni las 50 imágenes planas.

B. Falsificación de la Identidad de una Imagen

Para este experimento se utilizaron el mismo conjunto de fotografías que en el experimento de la eliminación del ruido de la sección VI-A. De forma similar al experimento anterior, se extrajo la huella de una de las cámaras y se inyectó a las otras dos haciendo uso del algoritmo propuesto y “PRNU Decompare”; después se compararon los resultados con la herramienta “NFI PRNU Compare”. Los roles que jugaron cada una de las cámaras se muestran en la Tabla II.

Tabla II
DISPOSITIVOS USADOS PARA LA FALSIFICACIÓN DE LA IDENTIDAD

Cámara Suplantadora	Víctima 1	Víctima 2
LG E510f	LG-400	Samsung GT-I8160P

Para realizar la falsificación del patrón del ruido del sensor con el algoritmo propuesto en este trabajo únicamente se requirieron las 50 imágenes planas uniformemente iluminadas pertenecientes a la cámara suplantadora. En el caso de la herramienta “PRNU Decompare”, para realizar la falsificación el programa requiere como entrada las 50 fotografías planas y la fotografía totalmente oscura tanto de la cámara suplantadora como de la cámara víctima.

Después de realizar la falsificación en las dos cámaras víctimas se compararon los resultados que están resumidos en la Tabla III. En el caso de la víctima 1 se puede observar que las dos suplantaciones resultaron tener mayor similitud con el patrón de la cámara suplantadora y el resultado de “PRNU Decompare” se acerca más, aunque la diferencia no es muy significativa considerando que ellos utilizan un número mucho mayor de imágenes como fuente de información.

En el caso de la víctima 2 el resultado del algoritmo propuesto fue el que menos similitud tuvo con el patrón de la cámara suplantadora. Los resultados obtenidos hasta el momento eran esperados, debido a que el algoritmo propuesto en este trabajo no asume que se tiene acceso a la cámara fuente y en el trabajo de “PRNU De compare” sí. Es importante notar que en escenarios reales normalmente no se tiene acceso a la cámara víctima.

VII. CONCLUSIONES

Se han presentado dos algoritmos, uno que permite destruir la identidad de una imagen y otro que posibilita falsificar la misma en una imagen dada. Los dos algoritmos tienen como base la utilización del ruido del sensor y la transformada wavelet.

Asimismo, ambos algoritmos tienen como gran ventaja con respecto a otros con los mismos fines que necesitan una menor variedad de datos de entrada ajustándose en mayor medida a escenarios reales. Concretamente, para el algoritmo de eliminación de la huella de una imagen se necesita únicamente la propia imagen y no un conjunto de imágenes planas y una imagen del marco oscuro de la propia cámara como ocurre en el caso de “PRNU Decompare”.

La aplicación a la realidad del algoritmo que utiliza “PRNU Decompare” no tiene gran facilidad de encaje, ya que se necesitan numerosas fotos con características concretas para su ejecución. Para el caso del algoritmo de falsificación de la identidad de una imagen propuesto no se necesita tener acceso a la cámara víctima.

Ambos algoritmos pueden verse desde otras perspectivas distintas a la del estudio para el futuro fortalecimiento de técnicas forenses de detección de manipulaciones intencionadas. Precisamente el primer algoritmo que permite destruir la identidad de una imagen puede ser de gran utilidad en aplicaciones web que presentan imágenes en Internet (redes sociales, directorios de imágenes, ...), ya que permiten que la imagen subida sea anónima desde el punto de vista de la identificación de la fuente de adquisición.

Tabla III
COMPARATIVA ENTRE PATRONES, IMÁGENES ORIGINALES Y VÍCTIMAS

Foto		Rojo	Verde	Azul	Suma
LG E510f		1	1	1	3
Víctima 1	Falsificada -Decompare	0,009219962	0,0054620425	0,009098741	0,023780745
	Falsificada -Propuesta	0,007900867	0,0046529872	0,0083521	0,020905953
	Original	0,0073570074	0,004183661	0,0075896666	0,019130334
Víctima 2	Falsificada -Decompare	0,01418404	0,013986045	0,013574668	0,041744754
	Original	0,011300047	0,013949845	0,0125216115	0,0377715
	Falsificada -Propuesta	0,008964902	0,0066337977	0,004440412	0,020039111

La eficacia de estos algoritmos están dentro de las expectativas esperadas, ya que aunque en algunos casos no obtienen resultados tan buenos como otras herramientas o algoritmos, si logran resultados cercanos y aceptables reduciendo drásticamente la diversidad y números de datos de entrada.

Estos algoritmos pueden ser de utilidad como punto de partida para futuras mejoras que permitan obtener resultados similares a los de otros algoritmos o herramientas, teniendo en cuenta como base inamovible la escasa variedad de datos de entrada necesarios y el no tener acceso a la cámara víctima en el caso de la falsificación de identidad de una imagen.

AGRADECIMIENTOS

Los autores agradecen el apoyo brindado por el "Programa de Financiación de Grupos de Investigación UCM validados de la Universidad Complutense de Madrid - Banco Santander".

REFERENCIAS

[1] M. Al-Zarouni, "Mobile Handset Forensic Evidence: a Challenge for Law Enforcement," in *Proceedings of the 4th Australian Digital Forensics Conference*, Perth Western, Australia, December 2006, pp. 1–10.

[2] M. Chen, J. Fridrich, M. Goljan, and J. Lukas, "Determining Image Origin and Integrity Using Sensor Noise," *IEEE Transactions on Information Forensics and Security*, vol. 3, no. 1, pp. 74–90, March 2008.

[3] L. J. García Villalba, A. L. Sandoval Orozco, and J. Rosales Corripio, "Smartphone Image Clustering," *Expert Systems with Applications*, vol. 42, no. 4, pp. 1927–1940, 2015.

[4] T. Van Lanh, K. S. Chong, S. Emmanuel, and M. S. Kankanhalli, "A Survey on Digital Camera Image Forensic Methods," in *Proceedings of the IEEE International Conference on Multimedia and Expo*, Beijing, China, July 2007, pp. 16–19.

[5] V. L. L. Thing, K. Y. Ng, and E. C. Chang, "Live Memory Forensics of Mobile Phones," *Digital Investigation*, vol. 7, pp. 74–82, August 2010.

[6] S. Bayram, H. T. Sencar, and N. Memon, "Classification of Digital Camera-Models Based on Demosaicing Artifacts," *Digital Investigation*, vol. 5, no. 1-2, pp. 49–59, September 2008.

[7] B. Wang, Y. Guo, X. Kong, and F. Meng, "Source Camera Identification Forensics Based on Wavelet Features," in *Proceedings of the International Conference on Intelligent Information Hiding and Multimedia Signal Processing*, Kyoto, Japan, September 2009, pp. 702–705.

[8] A. L. Sandoval Orozco, J. Rosales Corripio, L. J. García Villalba, and J. C. Hernández Castro, "Image Source Acquisition Identification of Mobile Devices based on the Use of Features," *Multimedia Tools and Applications*, pp. 1–25, 2015.

[9] A. L. Sandoval Orozco, D. M. Arenas González, J. Rosales Corripio, L. J. García Villalba, and J. C. Hernández Castro, "Source Identification for Mobile Devices, Based on Wavelet Transforms Combined with Sensor Imperfections," *Computing*, vol. 96, no. 9, pp. 829–841, September 2014.

[10] A. L. Sandoval Orozco, L. J. García Villalba, D. M. Arenas González, J. Rosales Corripio, J. C. Hernandez-Castro, and S. J. Gibson, "Smartphone Image Acquisition Forensics using Sensor Fingerprint," *IET Computer Vision*, June 2015.

[11] Z. J. Geradts, J. Bijhold, M. Kieft, K. Kurosawa, K. Kuroki, and N. Saitoh, "Methods for Identification of Images Acquired with Digital Cameras," in *Proceedings of the Enabling Technologies for Law Enforcement and Security*, vol. 4232, Boston, Massachusetts, USA, February 2001, pp. 505–512.

[12] J. Lukas, J. Fridrich, and M. Goljan, "Digital Camera Identification from Sensor Pattern Noise," *IEEE Transactions on Information Forensics and Security*, vol. 1, no. 2, pp. 205–214, June 2006.

[13] F. D. O. Costa, M. Eckmann, W. J. Scheirer, and A. Rocha, "Open Set Source Camera Attribution," in *Proceedings of the 25th Conference on Graphics, Patterns and Images*, Ouro Preto, Brazil, August 2012, pp. 71–78.

[14] T. Gloe, M. Kirchner, A. Winkler, and R. Bohme, "Can We Trust Digital Image Forensics?" in *Proceedings of the 15th International Conference on Multimedia*, Augsburg, Germany, September 2007, pp. 78–86.

[15] A. C. Popescu and H. Farid, "Exposing Digital Forgeries by Detecting Traces of Resampling," *IEEE Transactions on Signal Processing*, vol. 53, no. 2, pp. 758–767, February 2005.

[16] J. Lukas and J. Fridrich, "Estimation of Primary Quantization Matrix in Double Compressed JPEG Images," in *Proceedings of the Digital Forensic Research Workshop*, Cleveland, Ohio, August 2003, pp. 5–8.

[17] W. Chen, Y. Q. Shi, and W. Su, "Image Splicing Detection Using 2-D Phase Congruency and Statistical Moments of Characteristic Function," in *Proceedings of the Security, Steganography, and Watermarking of Multimedia Contents IX*, vol. 6505, San Jose, CA, January 2007, p. 26.

[18] M. Steinebach, H. Liu, P. Fan, and S. Katzenbeisser, "Cell Phone Camera Ballistics: Attacks and Countermeasures," in *Proceedings of the Multimedia on Mobile Devices*, San Jose, California, January 2010, pp. 75 420B–75 420B.

[19] "PRNU Decompare," http://sourceforge.net/projects/prnudecompare/.

[20] "NFI PRNU Compare," ftp://ftp.mirrorservice.org/sites/downloads.sourceforge.net/p/pr/prnucompare/PRNUCompare_bin_075.rar.

Jocelin Rosales Corripio received a Computer Science Engineering degree from the Benemérita Universidad Autónoma de Puebla (Mexico) in 2008. She holds a M.Sc. in Research in Computer Science from the Universidad Complutense de Madrid (Spain) in 2013. She is currently a Ph.D. student at the Universidad Complutense de Madrid (Spain) and a Research Assistant at Complutense Research Group GASS. Her main research interests are image processing and multimedia forensics.

Anissa El-Khattabi received a Computer Science Engineering degree from the Universidad de Granada (Spain) in 2009. She has worked as R&D Engineer in different companies. She is currently a Ph.D. student at the Universidad Complutense de Madrid (Spain) and a Research Assistant at Complutense Research Group GASS, which is located in the Faculty of Computer Science and Engineering at the UCM Campus. Her main research interests are image processing and multimedia forensics.

Ana Lucila Sandoval Orozco received a Computer Science Engineering degree from the Universidad Autónoma del Caribe (Colombia) in 2001. She holds a Specialization Course in Computer Networks (2006) from the Universidad del Norte (Colombia) and holds a M. S. in Research in Computer Science (2009) and a Ph.D. in Computer Science (2014), both from the Universidad Complutense de Madrid (Spain). She is currently a post-doc at Complutense Research Group GASS. Her main research interests are information security and its applications.

Luis Javier García Villalba received a Telecommunication Engineering degree from the Universidad de Málaga (Spain) in 1993 and holds a M.Sc. in Computer Networks (1996) and a Ph.D. in Computer Science (1999), both from the Universidad Politécnica de Madrid (Spain). Visiting Scholar at COSIC (Computer Security and Industrial Cryptography, Department of Electrical Engineering, Faculty of Engineering, Katholieke Universiteit Leuven, Belgium) in 2000 and Visiting Scientist at IBM Research Division (IBM Almaden Research Center, San Jose, CA, USA) in 2001 and 2002, he is currently Associate Professor of the Department of Software Engineering and Artificial Intelligence at the Universidad Complutense de Madrid (UCM) and Head of Complutense Research Group GASS (Group of Analysis, Security and Systems, http://gass.ucm.es) which is located in the Faculty of Computer Science and Engineering at the UCM Campus. His professional experience includes projects with Hitachi, IBM, Nokia, Safelayer Secure Communications and H2020. His main research interests are cryptography, coding, information security and its applications.

Multimed Tools Appl
DOI 10.1007/s11042-015-2633-2

Image source acquisition identification of mobile devices based on the use of features

Ana Lucila Sandoval Orozco[1] · Jocelin Rosales Corripio[1] · Luis Javier García Villalba[1] · Julio César Hernández Castro[2]

Received: 1 August 2014 / Revised: 25 February 2015 / Accepted: 13 April 2015

Abstract Nowadays, forensic analysis of digital images is especially important, given the frequent use of digital cameras in mobile devices. The identification of the device type or the make and model of image source are two important branches of forensic analysis of digital images. In this paper we have addressed both of these, with an approach based on different types of image features and the classification using support vector machines. The study has mainly focused on images created with mobile devices and as a result, the techniques and features have been adapted or created for this purpose. There have been a total of 36 experiments classified into 5 sets, in order to test different configurations of the techniques. In the configuration of the experiments, the future use of the technique by the forensic analyst in real situations to create experiments with high technical requirements was taken into account, amongst other things.

✉ Luis Javier García Villalba
javiergv@fdi.ucm.es

Ana Lucila Sandoval Orozco
asandoval@fdi.ucm.es

Jocelin Rosales Corripio
jocelinr@ucm.es

Julio César Hernández Castro
J.C.Hernandez-Castro@kent.ac.uk

[1] Group of Analysis, Security and Systems (GASS),
Department of Software Engineering and Artificial Intelligence (DISIA),
Faculty of Computer Science and Engineering, Office 431, Universidad Complutense de Madrid (UCM), Calle Profesor José García Santesmases, 9, Ciudad Universitaria, 28040 Madrid, Spain

[2] School of Computing, Office S129A, University of Kent, Cornwallis South Building, Canterbury CT2 7NF, UK

Published online: 14 May 2015

Keywords Forensics analysis · Counter forensics · Image anonymity · Image forgery · Photo response non uniformity · Wavelet

1 Introduction

The current demand for mobile devices (mobile phones, smartphones, tablets, etc.) is increasing year by year despite the global economic crisis. According to Gartner [17] in 2013 smartphone sales grew 42.3 % from the previous year; outnumbering the sales of feature phones for the first time. In total, according to estimates by the International Communication Union (ITU), there are 6.8 billion mobile phone subscriptions worldwide, which is a large increase on the 6 billion subscriptions in 2012 and 5.8 billion in 2011.

Having used figures to describe and illustrate the extent mobile devices are used across the world, we must not overlook the importance of having such devices available in our day to day lives. Increasing storage capacity, usability, portability and affordability, have allowed mobile devices to be present in several activities, places and events of daily life. So much so, that according to Ahonen and Moore [1], a large number of people have and use more than one mobile device and a typical user turns to their mobile devices an average of 150 times a day.

The majority of these mobile devices have an integrated digital camera, which in contrast to Digital Still Camera (DSC) are carried by their owners all the time to most places they attend. According to IC Insights, Inc. [23], the share of digital camera sales in mobile phones will be 48 %, with the share DCS sales in this year only 27 %. There are also predictions that the DSCs may disappear in favor of new integrated mobile device cameras [3], because the improved quality of these devices is growing at an unstoppable rate.

Because extensive use of mobile device digital cameras has generated controversy, discussions and rules have been made for the prohibition of using them in places such as government offices, schools and businesses, etc. A consequence of its widespread use, is that digital images can be used as silent witnesses in judicial proceedings (child pornography, industrial espionage, ...), and in many cases crucial pieces of evidence in a crime [2]. For these reasons, nowadays, digital image forensic analysis of mobile devices is very important.

Forensic analysis of digital images can be mainly divided into two branches [19]: tamper detection and image source identification. The first of the branches try to discern if an image has suffered any kind of processing after its creation, that is, the image has not been manipulated. The second branch will be presented in this work and it has the aim of identifying the type (camera, scanner, computer) or class (make and model) of the image source acquisition.

To understanding digital images forensic it is essential to know in detail the image acquisition process in digital cameras and scanners. Although many details of the digital camera pipeline belong to each manufacturer (and are maintained as confidential information) and specific device type, there is a general structure which is similar for all of them. In broad terms a digital camera consists of a system lens, a group of filters, an array of Color Filter Array (CFA), an image sensor and a Digital Image Processor (DIP) [5]. Van Lanh et al. [46] describe in detail the digital camera pipeline. The sensor is the component which is responsible for capturing light and creating a digital signal in terms of its intensity. Khanna et al. [25] describe the scanner pipeline, which is very different from the cameras. Graphically in the Fig. 1 a summary of it is showed.

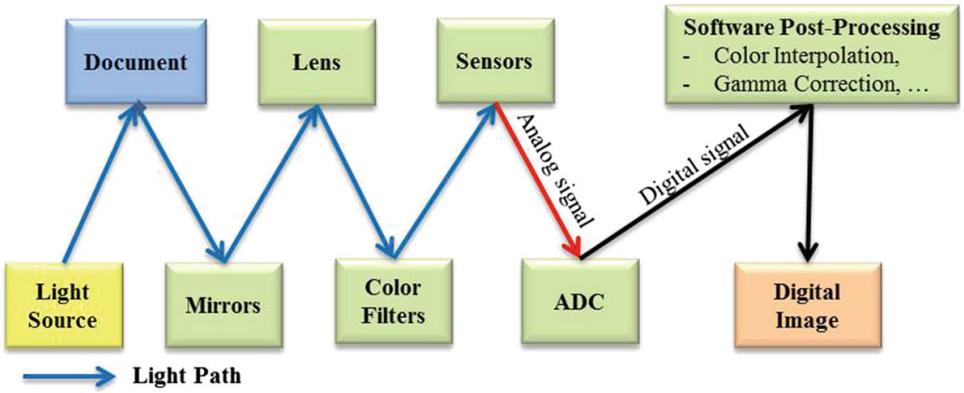


Fig. 1 Image acquisition pipeline in scanners

This paper is structured into 5 sections, with this introduction being the first. The rest of the paper is structured as follows. Section 2 carries out an analysis of techniques and algorithms for identifying the source type and source acquisition identification. Moreover a comparative table shows a summary of the findings. Section 3 shows different sets of features (Noise, Color, Image Quality Metrics (IQM) and Wavelets) used by the algorithms and techniques of forensic analysis. In Section 4, a set of experiments for the identification of device type and the source acquisition identification of the image are performed. In these experiments we use the set of the features previously presented and the algorithms of the techniques. Finally, Section 5 shows the main conclusions of this work and some potential lines of future work.

2 Forensic analysis techniques in digital images

In this section we describe the main techniques of digital image forensics for identifying the source of image acquisition and the main work of the analysis. Other compendiums of techniques may be shown in [36, 39, 42]. The success of these techniques depends on the assumption that all the images acquired by the same device have intrinsic features. The features which are used to identify the make and model of a digital camera are derived from the differences between the techniques of image processing technologies and the components which are used. The biggest problem with this approach is that different models of digital cameras use components of a small number of manufacturers, and the algorithms used are also very similar between models of the same brand. According to Van Lanh et al. [46] for this purpose four groups of techniques can be established depending on their base: lens system aberrations, Color Filter Array (CFA) interpolation, image characteristics, and sensor imperfections. In addition to the above there is another group of techniques based on metadata.

Techniques based on the image metadata use the information stored in the camera about the conditions of image capture in order to find information and image classification. The Exchangeable Image File Format(Exif) specification [14] is the most common container of metadata in digital cameras [3]. The Exif specification includes hundreds of labels, among which are *make* and *model*, although it should be noted that the specification does not make their existence in the image files compulsory. These techniques are the simplest. There are plenty of studies focused on the different types of metadata, both for finding information and for image classification [7, 35, 37, 43]. Metadata can also be used as input or aid for other

forensic techniques. For instance, in the application of content-based image techniques, Exif metadata can provide a large number and variety of technical information, which may allow an increase in the success rates or improve the results of the application of certain forensic algorithms [6, 16, 24]. However, these techniques depend largely in the metadata inserted by manufacturers when the image is created and the correction. Sandoval Orozco et al. [40] make an in depth study on this topic. Moreover, this method is the most vulnerable to malicious alterations.

Techniques based on lens aberration study several types of aberrations introduced by the lens system during the image generation process.

Choi et al. [13] propose the lens radial distortion as the best technique for source identification. The authors conclude that each camera model expresses a unique pattern of radial distortion which helps to uniquely identify it.

Lanh Tran Van et al. [45] propose lateral chromatic aberration as a technique for identifying source camera. The authors performed experiments using little sets of cameras with non-modified images or modified images with random crops regions. It was concluded that this technique is not suitable for identifying the source of different camera models from the same brand.

Some authors consider that the choice of the CFA matrix and the specification of color interpolation algorithms produce some of the most significant differences between different camera models [4, 5, 9, 28].

Long et al. [28] use correlations between pixels for the source identification. Neuronal networks are used for classification. The method was tested with cartoon images from four cameras. Since the cameras from the same manufacturer use the same color interpolation algorithm, this approach is not efficient at differentiating between different models from the same maker. Also, as shown in the experiments, good results are not obtained when the images have been modified or when they have high compression level.

Celiktutan et al. [9] use a set of binary similarity measures as metrics to estimate the similarity between image bit planes. This work uses a set of 108 binary similarity measures. The results of the method depends on the number of cameras used in the experiments.

Bayram et al. [5] present an algorithm for identifying and classifying color interpolation operations. This proposal is based on two methods to perform the classification process: first using an algorithm to analyze the correlation of each pixel value with its neighbors' values, and secondly an analysis of the differences between pixels independently. Different experiments with different numbers of cameras and image types were performed.

Cao and Kot [8] present a technique for source identification based on the information of the CFA matrix interpolation process and a comparison with other techniques. This technique has three new sets of demosaicing features: weights, Error Cumulants (EC) and Normalized Group Sizes (NGS). Since the number of features is very high a process (Eigenfeature Regularization (ERE)) is performed to decrease the number of it.

Ho et al. [20] propose four algorithms which are based on aspects of inter-channel correlation. These algorithms calculate variance maps (v-maps). The experiments image source identification uses four cameras for three different manufacturers and 50 images of each camera (25 for training and 25 test). The authors conclude that the inter-channel correlation provides a complementary approach to previous studies which dealing with correlations between pixels introduced in the demosaicing process.

Techniques based on image features use a set of features extracted from the content of the image to identify the source.

Tsai et al. [44] propose a method to identify the source using the following features: color features, image quality metrics and frequency domain. The study adopted the wavelet transforms as a method to calculate the wavelet domain statistics and use a Support Vector Machine (SVM) for classification. In experiments digital cameras and mobile devices were used. Tsai et al. [31] extend the source identification to different devices such as mobiles, phones, digital cameras, scanners and computers. In this proposal they base it on the differences in the image acquisition process to create two features groups: color interpolation coefficients and noise features. In the experiments they use five smartphone models, five digital camera models and four scanner models to identify the source type.

Wang et al. [47] propose a method for source camera identification based on the extraction and classification of wavelet statistical features. Finally 216 first-order wavelet features and 135 second order co-occurrence features is obtained. The most representative features are selected using an Sequential Forward Featured Selection (SFFS) algorithm and they are classified using a SVM.

Hu et al. [22] perform experiments with common imaging features to identify the source: wavelet, color, IQM, statistical features of difference images and statistical features of prediction errors. In the experiments, different combinations of different types of features are used and a SVM for classification of different devices with 300 images from each camera (150 for training and 150 for testing) and a resolution of 1024x1024 is used. Moreover experiments were performed to check the robustness against three of the most common alterations in digital images: JPEG compression, cropping and scaling. The final conclusions of this work are that some of the feature sets provide good success rates for intact images, but not for images with modifications. It also shows that different types of manipulations have different effects on success rates of different feature sets.

Ozparlak et al. [34] propose a technique for image source identification using ridgelets and contourlets subbands statistical models. After the feature extraction a SFFS algorithm is used for feature election. The method based on 216 wavelet features is considered useful only for the representation of a dimension, the approach based on ridgelets uses 48 features, and the approach based on contourlets includes a total of 768 features. The contourlets and ridgelets are not only effective in differentiating between camera models, but also to differentiate between natural images or those produced by computer, or to differentiate between images from scanners of the same maker. However the authors believe that improvements could be implemented experimenting with different selection algorithms.

Liu et al. [27] propose a method using the marginal density Discrete Cosine Transform (DCT) coefficients in low-frequency coordinates and neighboring joint density features from the DCT domain. Furthermore, hierarchical clustering and SVM is used to detect the source of acquisition of the images.

Sandoval et al. [41] propose the mixture of two techniques (Sensor Imperfections and Wavelet Transforms) to get the source identification of images generated with mobile devices. This method extracts the sensor noise patterns of images, and then, a set of 25 features are obtained (16 first-order and higher-order features and 9 features by applying QMFs (Separable Quadratic Mirror Filters)).

Techniques based on Sensor imperfections study the fingerprints which can leave sensor defects on pictures. These techniques are divided into two branches: pixel defects and Sensor Pattern Noise (SPN). In the first pixel defects, hot pixels, dead pixels, row or column defects and group defects are studied. In the second pattern noise by averaging multiple noise residuals obtained by any noise removal filter is constructed.

The presence of the pattern is determined using a classification method as correlation or SVM.

Geradts et al. [18] study pixel defects of Charge Coupled Device (CCD), sensors, focusing on different features to analyze images and then identify their source: CCD sensor defects, the file format used, noise introduced in the image and watermarking introduced by makers. Among the CCD sensor defects are considered hot spots, dead pixels, group defects, and row or column defects. Results indicate that each camera has a different defect pattern. Nevertheless, it is also noted that the number of pixel defects for images from the same camera is different and varies greatly depending in the image content. Likewise, it was revealed that the number of defects varies with temperature. Finally, the study found that high quality CCD cameras do not have this kind of problem. When considering only defective CCD sensors this study is not applicable to the analysis of images generated by mobile devices.

Lukas et al. [29] analyze the sensor pattern noise from a set of cameras, which functions as a fingerprint allowing the unique identification of each camera. To identify the camera from a given image, the reference pattern is considered as a watermark in the image and its presence is established by a correlation detector. This method is affected by processing algorithms such as image Joint Photographic Experts Group (JPEG) compression and gamma correction. According to Van Lanh et al. [46] the results for pictures with different sizes were unsatisfactory. Also in this technique, images whose reference pattern is extracted must have the same size as the test images.

Costa et al. [15] propose an approach to source camera identification in open set scenarios, where unlike closed scenarios it is not assumed to have access to all the possible image source cameras. This proposal includes three phases: definition of regions of interest, determining the characteristics and source camera identification. Different regions of the images can contain different information about the fingerprint of the source camera. Besides, this approach in contrast to others considers 9 different Region Of Interests (ROI), not only the central region of the image. Using these ROIs it is possible to work with different resolution. For determining the features the SPN for each of the R, G, B and Y (luminance) channels is calculated, generating a total of 36 representative features for each image. Costa et al. [33] extended this approach, where in addition to presenting other techniques and algorithms, new experiments are performed. In experiments 13210 images of 400 cameras were used (they only have physical access to 25 cameras, the rest are images downloaded from *Flickr*) and they obtained better success rates. of 96.56 %, 97.34 %, 96.80 % and 97.18 %, using open sets with 2/25, 5/25, 10 /25 and 15/25 cameras respectively.

Table 1 shows a summary of the findings described above. The no detailed information in articles has been filled with (ND). We must take into account that in most of the above articles various experiments with different numbers of cameras and images are performed. In the column "Number of models/makers" the total models and manufacturers used for all experiments are accounted, which does not imply that in all experiments all models from all manufacturers are used. The "Applied to mobile devices" column indicates that at least one of the models used in the experiments is a mobile device. The column "Applied to different models of the same brand" indicates that at least in one experiment cameras from the same manufacturer were used. In each experiment an average success rate is obtained in the source identification, the "Minimum and maximum success rate" column shows the minimum and maximum values for the different experiments (in the case that there is only one value is because this article only has one experiment).

Table 1 Evaluation of camera identification techniques

Technique	Proposal	Classification Method	Number of Models / Makers	Image formats	Resolution	Applied to mobile device	Applied to different models of the same brand	Minimum and maximum success rate
Lens Aberration	[13]	SVM	3/3	JPEG	Different	ND	No	87.38 %−91.53 %
	[45]	SVM	3/3	JPEG	Different	No	Yes	72.75 %−92.22 %
CFA Interpolation	[28]	Neuronal Network	4/4	Uncompressed	ND	No	No	98.25 %
	[9]	SVM Linear and Non-linear RBF	9/3	ND	Different	Yes	Yes	62.3 %−98.7 %
	[5]	SVM Linear and Non-linear RBF	5/5	JPEG	Different	No	Yes	84.8 %−88 %
	[8]	PSVM and 1NN	4/11	JPEG	Different	Yes	Yes	94.8 %−99.4 %
	[20]	1NN	4/3	JPEG	ND	No	Yes	94.5 %
Image Features	[44]	SVM Linear	2/7	JPEG	1600x1200	Yes	Yes	61.7 %−99.72 %
	[31]	SVM Linear	5/5	JPEG	ND	Yes	No	97.7 %
	[47]	SVM Non-linear RBF	6/4	JPEG	ND	No	Yes	98 %
	[22]	SVM Non-linear RBF	10/4	JPEG	1024x1024	No	Yes	47 %−92 %
	[34]	SVM Non-linear RBF	3/3	ND	ND	No	No	93.33−99.7 %
	[27]	SVM Linear and Non-linear RBF	5/4	JPEG	Different	Yes	Yes	86.36 %−99.91 %
	[41]	SVM	10/6	JPEG	ND	Yes	Yes	89.45 %
Sensor Imperfections	[18]	ND	2/2	ND	640x480	No	No	ND
	[29]	ND	5/9	Different	Different	No	Yes	ND
	[15]	SVM Non-linear RBF	25/9	JPEG	Different	Yes	Yes	94.49 %−98.10 %
	[33]	SVM Non-linear RBF	25/9	JPEG	Different	Yes	Yes	96.56 %−97.34 %

3 Image source acquisition identification techniques

This section will propose techniques for image source acquisition identification (source type or source make and model) based on feature extraction from image content. There will be a set of images from known sources to be used for training an SVM classifier [21] and another set of images from unknown sources that will be used in the test stage to find out their acquisition source.

Our approach can be used to analyze images with different acquisition situations and resolutions, with successful identifications results. In addition, the source identification method that we propose is more general because it is useful in a larger set of classification problems. Our contributions to this paper, which improve the approach by Sandoval Orozco et al. [41] can be summarized as follows:

- A new features generation approach in which the following can be found: sensor pattern noise, color features, image quality metrics and wavelets features. The combination of these features allow the image source identification of images from different types of devices between (images from mobile phones, images obtained from a scanner, and a computer-generated images) and mobile devices of the same brand and different model.
- Two new algorithms: one to extract the sensor pattern noise and another to obtain the features of the fingerprints obtained.

Regarding classification, Michie et al. [32] perform a study of different classification methods such as distance-based classifiers, Bayesian classifiers, neural networks, clustering algorithms and SVM classifiers. As can be observed in the review, the use of SVM classifiers is widely used for these purposes. The kernel choice depends, among other factors, on the nature of the data to be classified. This paper will use an SVM classifier with Non-linear RBF kernel, as it is recommended for use when there is no a priori information about the data. The parameters for the SVM are the same as those used in [38]. Likewise, the option chosen is the most widely used one by the most recent precise works and they present good results. There are many implementations of SVM classifiers; particularly in this work we opted to use the LibSVM library [11].

The set of features to be used can be classified into four major groups, depending on the nature of their obtaining: noise features (16 features), color features (12 features), IQM (40 features) and wavelets (81 features). A detailed analysis on each of the aforementioned feature sets will be performed below.

3.1 Noise features

The image generation process tends to introduce various defects in them, which will create noise that will be shown in the final image. One type of noise is caused by defects in the CFA matrix, which include hot point defects, dead pixels, pixel traps, column defects and cluster defects. Such defects cause said pixels to differ largely from the rest in the original image; in many cases it makes no difference to have one image or another, since this pixel will always show the same value. For example, dead pixels will appear in the image as black pixels, or hot point pixels will appear as very bright pixels. The noise pattern in an image refers to any spatial pattern that does not change from one image to another and is caused by a “dark current” and a (PRNU) [26]. There are several filters to soften the effect of this noise. The Gaussian filter will be used for simplicity

and speed. This filter will be used to eliminate noise in images and then obtain different features.

One of the objectives is to get a set of features that allow us to differentiate between the different types of devices. To do this we firstly take into account that digital cameras use a two-dimensional array sensor whereas most scanners use a linear array sensor. In the case of scanners, the linear arrangement of the sensor moves to generate the entire image, so it is expected to find the periodicity of the sensor noise within the rows of the scanned image. On the other hand, there is no reason to find sensor noise periodicity within the columns of the scanned image. In the case of digital cameras this type of noise periodicity does not exist. This difference can be used as a basis to discriminate between different types of devices. Noise features extraction is based on [25].

Let I an image of $M * N$ pixels, M as the rows and N as the columns. We denote I_{noise} the noise of the original image and I_{noise} is the image without noise. Therefore:

$$I_{noise} = I - I_{denoised}$$

Then, each color component of the image without noise is subtracted to each color component of the original image, with which we obtain noise components of each pixel disaggregated for each color component.

The image original noise I_{noise} can be modeled as the sum of two components, the constant noise $I_{noiseconstant}$ and random noise $I_{noiserandom}$. For scanners constant noise only depends of the column index, because the same sensor is moved vertically to generate the complete image. The average noise of all columns can be used as a pattern reference $\hat{I}_{noiseconstant}(1, j)$ because the random noise components were cancelled.

$$\hat{I}_{noiseconstant}(1, j) = \frac{\sum_{i=1}^{M} I_{noise}(i, j)}{M}, 1 \leq j \leq N$$

For detecting the similarity between different rows with the pattern reference, we use the correlation of these rows with the pattern.

$$correlation(X, Y) = \frac{(X - \overline{X}) \cdot (Y - \overline{Y})}{\|X - \overline{X}\| \cdot \|Y - \overline{Y}\|}$$

Then the same process is performed to detect the similarity of the columns with the pattern reference. After obtaining the correlation between rows and between columns we will go to obtain the feature set. It should be noted at the time of obtaining the features, that in the case of scanners the orientation of the image is critical, because features obtained will be completely different.

For each type of correlation first order statistical values are obtained, which are: mean, median, maximum and minimum. Mode feature was discarded, since after several analysis and experiments was observed it was a useless feature, because when we are dealing with floating values, they did not exist in the majority of the cases repeated values. Tests were performed truncating float values, but the results were not good, decreasing the success rate. Other high order features are variance, kurtosis and skewness. All of them measure

statistical values more specifically than previous ones. Also, the ratio features between rows and columns correlations are added. Finally the average noise per pixel feature was included. This feature does not depend on rows or columns correlations with the reference pattern, but is independent and it can distinguish between different types of devices, such as computer generated images.

In total a set of 16 features are obtained: 7 rows features, 7 columns features, the ratio between rows and columns correlations and the average noise per pixel.

3.2 Color features

The configuration of the CFA filters, the demosaicing algorithm and color processing techniques mean that signals in the color bands may contain treatments and specific patterns. In order to determine the differences in color features for different camera models, it is necessary to examine the first and second order statistics of the pictures taken with them. Then, a set of 12 color features based [22] are proposed.

- **Pixels average value**. For this measure it is assumed that the average values of the RGB channels of an image should be the gray color, as long as the image has enough color variations. This measure is performed for each RGB channels (3 features).
- **Correlation pair between RGB bands**. This measure expresses the fact that depending on the structure of the camera, the correlation between the different color bands can change. In the implementation of this feature it uses the Pearson correlation coefficient to determine the correlation values between the bands. As a result we obtain three features which come from measuring the correlation between the RG, RB and GB bands.
- **Neighbor distribution center of mass for each color band**: This measure is calculated for each band separately (3 features). Firstly, the total number of pixels for each color value is calculated, obtaining a vector with 256 components. Then, with these calculated values the sum of neighboring values are obtained, that is, for each i value of the vector previously calculated the component $i - 1$ and $i + 1$ is added. Finally, the center of mass of the latter vector is calculated, which will return a value between 0 and 255.
- **Energy ratios between pairs RGB**. This feature depends on the white dots correction process of the camera. They are 3 features which are defined as:

$$E_1 = \frac{|G|^2}{|B|^2} \quad E_2 = \frac{|G|^2}{|R|^2} \quad E_3 = \frac{|B|^2}{|R|^2}$$

3.3 Image quality metrics (IQM)

Different camera models produce images of different quality. There may be differences in image brightness, sharpness or quality color. These differences propose a set of quality metrics features that help us to distinguish the image source. There are different IQM categories: measures based on the pixels differences, measures based on correlation and

measures based on spectral distance. For obtaining this set of metrics, a filtered image in which the noise of the original image is reduced to perform different calculations is needed in addition to the original image. For this, a Gaussian filter that allows us to perform image smoothing is used. For obtaining two-dimensional Gaussian kernel we used:

$$\left(2\pi\sigma^2\right)^{-1} * e^{\frac{-\left(i^2+j^2\right)}{2\sigma^2}}$$

After the core is obtained, it is normalized, so that the sum of all its components is 1. This is necessary to obtain a smooth image but with the same colors as the original. The normalization is performed dividing each component by the sum of the values of all the components. For obtaining the metrics a filter with a 3x3 kernel with $\sigma = 0.5$ is used:

$$h = \begin{matrix} 0.01134 & 0.08381 & 0.01134 \\ 0.08381 & 0.61934 & 0.08381 \\ 0.01134 & 0.08381 & 0.01134 \end{matrix}$$

Each pixel of the new image is obtained by transforming the pixel neighborhood over the original pixel image using the kernel previously calculated.

$$\begin{aligned} I'(x, y) = {} & h(0,0) * I(x-1, y-1) + h(0,1) * I(x, y-1) + h(0,2) * I(x+1, y-1) + \\ & h(1,0) * I(x-1, y) + h(1,1) * I(x, y) + h(1,2) * I(x+1, y) \\ & h(2,0) * I(x-1, y+1) + h(2,1) * I(x, y+1) + h(2,2) * I(x+1, y+1) \end{aligned}$$

It is necessary to consider the edges of the image to make the transformation. In our case we consider an outer edge with pixel value 0.

Following the specification of the 40 IQM features based on [22].

– **Czekonowsky distance**: The Czekonowsky distance is a useful metric for comparing vectors with no negative components as in the case of color images.

$$M = \frac{1}{MN} \sum_{i=1}^{M} \sum_{j=1}^{N} \left(1 - \frac{2 \sum_{k=1}^{3} min\left(C_k(i, j), \hat{C}_k(i, j)\right)}{\sum_{k=1}^{3} \left(C_k(i, j) + \hat{C}_k(i, j)\right)} \right)$$

In this formula and the subsequent $C_k(i, j)$ and $\hat{C}(i, j)$ refer to the pixels in the (m, n) position of the original image and the smoothed image (filtered image with noise reduction) respectively. Furthermore, M and N are the horizontal and vertical size of the image respectively.

– **Minkowsky metrics**: Minkowsky metrics for $\gamma = 1$ and $\gamma = 2$ are based on formula (1).

$$M_\gamma = \frac{1}{K} \sum_{k=1}^{K} \left\{ \frac{1}{N^2} \sum_{i,j=1}^{N} \left| C_k(i, j) - \hat{C}_k(i, j) \right|^\gamma \right\}^{\frac{1}{\gamma}} \tag{1}$$

This formula calculates the norm L_γ of dissimilarity between two images, where N^2 is the total number of pixels. In this formula and ongoing, k will refer to each of the image channels. It must be taken into account that this formula performs the average of Minkowski metric for all channels of the image.

$\gamma = 1$ is corresponding with the Mean Absolute Error (MAE) and $\gamma = 2$ with the Mean Square Error (MSE). In both cases, high values of MAE or MSE correspond with low quality images.

- *Mean Absolute Error*:

$$MAE = \frac{1}{MN} \sum_{m=1}^{M} \sum_{n=1}^{N} \left| C_k\,(i,\,j) - \hat{C}_k(i,\,j) \right|$$

- *Mean Square Error*:

$$MSE = \frac{1}{MN} \sum_{m=1}^{M} \sum_{n=1}^{N} \left(C_k\,(i,\,j) - \hat{C}_k\,(i,\,j) \right)^2$$

These metrics are applied to each of the bands separately, so that three features for the MAE and others three for the MSE are obtained

- **Laplacian Mean Square Error (LMSE)**: This metric is based on the importance of measuring the edges. A LMSE high value indicates that image quality is poor. It is defined as follows:

$$LMSE = \frac{\sum_{m=1}^{M} \sum_{n=1}^{N} \left[L\,(x\,(m,n)) - L\left(x^{(m,n)}\right) \right]^2}{\sum_{m=1}^{M} \sum_{n=1}^{N} \left[L\,(x\,(m,n)) \right]^2}$$

where $L\,(x\,(m,n))$ is the Laplacian operator:

$$L\,(x\,(m,n)) = x\,(m+1,n) + x\,(m-1,n) + x\,(m,n+1) + x\,(m,n-1) - 4x\,(m,n)$$

- **Normalized Cross Correlation**: The closeness between two digital images can also be quantified in terms of a correlation function. The quality metric of the normalized cross-correlation measurement for each image band k is defined as:

$$NCC = \frac{\sum_{i,j=0}^{N-1} C_k\,(i,\,j) * \hat{C}_k\,(i,\,j)}{\sum_{i,j=0}^{N-1} C_k\,(i,\,j)^2}$$

- **Structural Content**: The structural content of an image quality metric is defined for each band k as:

$$SC = \frac{\sum_{i,j=0}^{N-1} C_k\,(i,\,j)^2}{\sum_{i,j=0}^{N-1} \hat{C}_k\,(i,\,j)^2}$$

– **Spectral Measures**: The Discrete Fourier Transform (DFT) of the original image and the smoothed image, denoted as $\tau_k(u, v)$ and $\hat{\tau}_k(u, v)$ for a band k, are defined respectively as:

$$\tau_k(u, v) = \sum_{m,n=0}^{N-1} C_k(m, n) * e^{\left[-2\pi i m \frac{u}{N}\right]} * e^{\left[-2\pi i n \frac{v}{N}\right]}, u = 0 \ldots M-1, v = 0 \ldots N-1$$

$$\hat{\tau}_k(u, v) = \sum_{m,n=0}^{N-1} \hat{C}_k(m, n) * e^{\left[-2\pi i m \frac{u}{N}\right]} * e^{\left[-2\pi i n \frac{v}{N}\right]}, u = 0 \ldots M-1, v = 0 \ldots N-1$$

Where (u, v) are the coordinates of an image pixel in transform domain. The phase and magnitude of the DFT spectrum are defined as:

$$\varphi(u, v) = \arctan(\tau_k(u, v))$$

$$M(u, v) = |\tau_k(u, v)|$$

With the above concepts the following image quality metrics can be defined for each image band:

– *Spectral Phase*:

$$SP = \frac{1}{MN} \sum_{u=1}^{M} \sum_{v=1}^{N} \left|\varphi(u, v) - \hat{\varphi}(u, v)\right|^2$$

– *Spectral Magnitude*:

$$SP = \frac{1}{MN} \sum_{u=1}^{M} \sum_{v=1}^{N} \left|M(u, v) - \hat{M}(u, v)\right|^2$$

– *Weighted Spectral Distance*: Performs a weighted average of the phase and magnitude spectrum: $WSD = \rho * SM + (1 - \rho) * SP$, where for our case $\rho = 2.5 * 10^{-5}$.

These characteristics can also be obtained for each image block. For this we consider that the image is divided in L blocks with bxb size, and then the above features are calculated. In this way the following l-th block features for each band of the block can be defined:

$$J^l{}_\varphi = \left(\sum_{u,v=0}^{b-1} \left(\varphi^l(u, v) - \hat{\varphi}^l(u, v)\right)^\gamma \right)^{\frac{1}{\gamma}}$$

$$J^l{}_M = \left(\sum_{u,v=0}^{b-1} \left(M^l(u, v) - \hat{M}^l(u, v)\right)^\gamma \right)^{\frac{1}{\gamma}}$$

$$J^l = \rho * J^l{}_M + (1 - \rho) * J^l{}_\varphi$$

For calculating these features we have used $\gamma = 2$ and a 32x32 block size. After calculating these measures for each block we can get the following features:

– *Median Block Spectral Magnitude*: MBSM = $median J^l{}_M$, l = 1…L

- *Median Block Spectral Phase*: MBSM = $median J^{l}{}_{\varphi}$, l = 1...L
- *Median Block Weighted Spectral Distance*: MBWSM = $median J^{l}$, l = 1...L

- **Measures based on the human visual system**: Images can be processed by filters which simulate the Human Visual System (HVS). One of the models used for this is a band-pass filter with a transference function in polar coordinates:

$$H(\rho) = \left\{ \begin{array}{cc} 0.05e^{\rho^{0.554}} & \rho < 7 \\ e^{-9[|log_{10}\rho - log_{10}9|]^{2.3}} & \rho \geq 7 \end{array} \right\}$$

where $\rho = \sqrt{\left(u^2 + v^2\right)}$. The operator U is defined as:

$$U\left\{C\left(i,\, j\right)\right\} = DCT^{-1}\left\{H\left(\sqrt{u^2 + v^2}\right)\omega\left(u,\, v\right)\right\}$$

where $\omega\left(u,\, v\right)$ denotes the two-dimensional DCT of the image and DCT^{-1} is the inverse two-dimensional DCT.

Finally the image quality metrics that we obtain for each band of the image based on these measures are:

- *Normalized absolute error*:

$$NAE = \frac{\sum_{i,j=0}^{N-1}\left|U\left\{C_k\left(i,\, j\right)\right\} - U\left\{\hat{C}_k\left(i,\, j\right)\right\}\right|}{\sum_{i,j=0}^{N-1}\left|U\left\{C_k\left(i,\, j\right)\right\}\right|}$$

- *HVS based L2*:

$$L2 = \left\{\frac{1}{N^2}\sum_{i,j=0}^{N-1}\left|U\left\{C_k\left(i,\, j\right)\right\} - U\left\{\hat{C}_k\left(i,\, j\right)\right\}\right|^2\right\}^{\frac{1}{2}}$$

3.4 Wavelet features

Due to the deterministic property of the sensor pattern noise which is present in an image, this pattern can be used as a footprint to identify the device that generated the image under investigation. It can be said that the sensor pattern noise is to a digital camera as a fingerprint is to a human being.

To identify the acquisition source we require an algorithm that allows us to extract the sensor noise and another that allows us to obtain the features of the fingerprints obtained in order to classify and identify them.

Taking the main ideas from Lukas et al. [29] as a reference, algorithm 1 is proposed to extract sensor noise.

Algorithm 1 Extracting PRNU

Input: Image I
Variance estimation: adaptive or non-adaptive
Result: Sensor fingerprint I_{noise}

1 **procedure** EXTRACTPRNU(I)
2 Apply a wavelet decomposition in 4 levels to I;
3 **foreach** *wavelet decomposition level* **do**
4 **foreach** *component* $c \in \{H, V, D\}$ **do**
5 Compute the local variance;
6 **if** *adaptive variance* **then**
7 Compute 4 variances with windows of size: 3, 5, 7 and 9 respectively;
8 Select the minimum variance;
else
9 Compute the variance with a window of size 3;
10 Compute noiseless wavelet components applying the Wiener filter to the variance;
11 Obtain I_{clean} by applying the inverse wavelet transform with clean components calculated;
12 Obtain the sensor noise with $I_{noise}=I-I_{clean}$;
13 Apply zero-meaning to I_{noise};
14 Increase the green channel weight with $I_{noise}= \quad 0.3 \cdot I_{noise_R} + 0.6 \cdot I_{noise_G} + 0.1 \cdot I_{noise_B}$;
15 **end procedure**

With averaging set to zero, the fingerprint is removed of features which are not intrinsic to the sensor, as Choi et al. [12] suggest, so that the averages of the rows and columns are equal to zero. This is achieved by subtracting the average of the column from each pixel in the column, and then by subtracting the average of the row from each pixel in the row. This operation is applied to all rows and columns of the image. After cleaning the image, the green channel is given greater weight since due to the color matrix settings it usually contains more information about the image than the rest of the color channels [10, 30]. Finally, a total of 81 features (3 channels x 3 wavelet components x 9 central moments) are calculated using algorithm 2.

Algorithm 2 Extracting features

Input: Sensor fingerprint I_{noise}
Result: 81 features

1 **procedure** EXTRACTFEATURES(I)
2 Separate R, G and B color channels of I_{noise};
3 **foreach** *color channel* **do**
4 Apply a wavelet decomposition in 1 level;
5 **foreach** *component* $c \in \{H, V, D\}$ **do**
6 Compute k central moments with $m_k = \frac{1}{n} \sum_{i=1}^{n} |c_i - \overline{c}|^k$;
7 **end procedure**

4 Experiments

This section shows the results of the experiments conducted to identify the type of source device and for source acquisition identification.

It should be noted that the classification of images to be performed in this work be done on what can be called a closed set of elements, i.e., the classes of the elements used in training are the same classes as those used in the test. The images used in the training stage are not used in the testing stage.

4.1 Source device type identification

In this experiment we will use an image set composed of: images from mobile phones, images obtained from a scanner, and a computer-generated images. 200 images are used from each set, 100 for the SVM training and 100 for testing. All images have a resolution higher than 1024x768. There is no restriction on the content of the image or the camera configuration parameters at the time of the acquisition.

Images from mobile phones have been obtained from known phones, so the origin of the source can be ensured. Images from 12 smartphones, some of them from the same manufacturer, were selected.

For images from scanners and computer-generated images, our own sources and the *Flickr* website were used. The set of images downloaded from the web had a set of filters applied to them in order to obtain a set with higher reliability that would introduce the least possible noise into the experiments. All images downloaded from *Flickr* are originals with no resizing. As a second filter for scanned images, those which had the tag "scanned images" and made reference to a retail scanner model were used. For computer-generated images, we discarded the images that had a "camera model" tag with a value from a retail scanner or camera. 15 scanned images were selected, some of which were chosen from the same manufacturer. With respect to computer-generated images, precise information on the number of applications or type of computers used cannot be indicated.

As can be seen, there is a high number of different kinds of devices (makes and models) of the three types, which greatly hampers the classification.

For the experiments we have taken into account the following configuration parameters: size of crop applied to the image, crop position (centered or upper-left corner) and application of different feature sets (Noise Features, Color Features, IQM Features and Wavelet Features).

Table 2 shows the results of success rates and the configuration parameters used in the 10 experiments.

From the analysis of the results, general and specific conclusions about the various configurations used in each experiment can be obtained. Encompassing all the experiments, it is observed that success rates are not excessively high (60.42 % on average and 71.30 % in the best case); it can be concluded that this technique is not particularly suitable for this purpose. It is important to emphasize, as noted above, that the number of different makes and models used for this experiment is high, which predictably causes success rates to drop. That being said, it should be noted that this study does provide interesting results on the configuration parameters used, since between the best and the worst result there is a difference in the average success rate of 23.48 %.

In general it can be concluded that the only use of the noise features do not perform well for identifying the source type when the number of devices to be classified is high, since the average success rate of all experiments is 56.65 % . Since the results are not good then a

Table 2 Source device identification between mobile camera, scanner and computer generated images

Features	Crop Size	Crop Align	Device			Average
			Camera	Computer	Scanner	
Noise	Full Size	–	70	54	57	59.95
Noise	1024x768	Center	66	80	46	62.39
Noise	800x600	Center	76	60	49	60.68
Noise	640x480	Center	62	61	48	56.62
Noise	1024x768	Upper-left corner	76	59	40	56.40
Noise	800x600	Upper-left corner	65	38	44	47.72
Noise	640x480	Upper-left corner	74	54	37	52.88
All Features	1024x768	Center	66	73	72	70.26
All Features	800x600	Center	69	74	71	71.30
All Features	640x480	Center	77	73	63	70.75
Average			69.91	61.35	51.42	60.42

set of experiments reducing the number and types of devices will be made to observe their results.

The results improve significantly when all the features to identify the source type are used. Given the high number of classes, the results can be qualified as acceptable, since the average success rate for all experiments carried out using these features is 70.77 %. Also, as can be observed in the results, we conclude that the crop size affects the results: the smaller the crop, the lower the success rate, even if the differences are not extremely significant. It is also noteworthy that with a 1024x768 crop size better results are obtained than when using the full-sized image, i.e., from a given crop size, the results get worse.

A series of experiments will be performed reducing the number of types to be classified, in order to test the behavior of the results when we only use noise features, since these are the ones that obtained the worst results previously. The results are shown in Table 3.

As can be seen, the success rate goes up considerably as it was expected, reaching 85.44 % on average. When the number of types of devices is reduced to two and as a result the number of classes is reduced the results are acceptable.

Table 3 Source type identification with noise features

Device Type 1	Device Type 2	Features	Crop Size	Crop Align	Average
Scanner	Smartphone	Noise	1024x768	Center	95.79
Scanner	Smartphone	Noise	640x480	Center	96.16
Scanner	Smartphone	Noise	400x300	Center	96.73
Computer	Smartphone	Noise	1024x768	Center	79.96
Computer	Smartphone	Noise	640x480	Center	79.76
Computer	Smartphone	Noise	400x300	Center	78.55
Computer	Scanner	Noise	1024x768	Center	82.87
Computer	Scanner	Noise	640x480	Center	81.10
Computer	Scanner	Noise	400x300	Center	80.91

The first general conclusion obtained corroborates an earlier conclusion, since it is observed that the crop size does not significantly affect the results. The best results are those that distinguish between smartphone and scanned images, with 96.23 % average success rate. The second best result appears with the distinction between scanned and computer-generated images, with 81.62 % average success rate. The worst result was obtained in the distinction between computer-generated and smartphone images, with 79.42 % average success rate. Still, any of the results of these experiments are significantly better than the results in Table 2. Therefore, it can be concluded that in general, the use of noise features for type of source distinction only obtains acceptable results when the number of classes is not high.

4.2 Image source identification for mobile devices

Given the importance of mobile images today, below we will show the experiments performed to identify the acquisition source of images from mobile devices, i.e., the classification of an image set according to the make and model of the camera that generated them.

In these experiments a set of 200 images will be used, 100 for the SVM training and 100 for testing. 12 smartphone models were used: IPhone 4s (I1), IPhone 5s (I2), Blackberry 8520 (BB), Huawei U8815 (HU), LG E400 (LG1), LG P760 (LG2), Nokia 800 (N1), Samsung GT-I9001 (S1), Samsung GT-I9100 (S2), Samsung GT-I8160P (S3), Samsung GT-5830M (S4) and Sony C2105 (SE1). The images comply with the same restrictions as the cameras in the previous section.

The experiments have been grouped into 3 groups with the aim of obtaining conclusions on: the use of different feature sets, crop size, the number of devices used for the classification, and the use of devices from the same manufacturer. The experiments where all devices are from the same manufacturer put the techniques presented to the test. Hardware and software components of the cameras from the same manufacturer are generally very similar or even the same, which obviously presents serious difficulties or impossibility of distinction among the different smartphone models.

Table 4 shows the first set of experiments in which 7 models of mobile devices from different manufacturers are used. Different types of combinations of features sets were tested. Most experiments were performed with a crop size of 1024x768, since as this is considered a large enough size to obtain good results, as shown in the previous experiments.

The experiment reveals that noise, color and IQM feature sets are individually completely invalid, since the best result obtains an 37.67 % average success rate, which is unacceptable. With the remaining set of features (wavelets), two experiments were conducted using different types of wavelet: Daubechies 8-tap and Haar. The results show that Daubechies 8-tap obtains better results than Haar and the best results of all experiments (91.46 %).

With respect to the different feature combinations, it is observed that when we use all the features good results are obtained (86.54 % in the best case), since, although they are slightly worse than the best result, the difference is not very significant (4.92 %). Also, the success rate when all the features are used subtly drops the smaller the crop size gets.

The combination of all the features except noise features, which are mainly focused on identifying the source type, yields an average success rate of 83.67 %. These results, even if not bad, are far from those obtained with the wavelets and worse than when the combination of all features is used.

Table 4 Image source acquisition identification for 7 smartphones

Features	Crop Size	Crop Align	I1	HU	LG2	N1	BB	S1	SE1	Average
All Features (Daubechies 8-tap)	1024x768	Center	93	96	80	94	91	70	85	86.54
Noise	1024x768	Center	41	42	35	18	40	40	62	37.67
Color	1024x768	Center	24	37	20	40	31	19	44	29.27
IQM	1024x768	Center	13	88	46	89	7	34	2	21.65
Wavelet Daubechies 8-tap	1024x768	Center	95	96	96	94	92	76	93	91.46
Wavelet Haar	1024x768	Center	95	87	97	70	86	56	91	81.84
Color + IQM + Wavelet Daubechies 8-tap	1024x768	Center	93	94	90	90	90	53	85	83.67
All Features(Daubechies 8-tap)	800x600	Center	91	96	84	92	95	56	85	84.41
All Features (Daubechies 8-tap)	640x480	Center	90	95	84	89	88	51	88	82.15

In order to further evaluate the results of using all feature sets, in the next set of experiments 10 models of mobile devices, some of them from the same manufacturer are used. The results of the experiments are shown in Table 5.

As previously stated, the fact that there are devices from the same manufacturer and similar features greatly hampers the classification task, since cameras can be identical or virtually identical. As expected, we find that the larger the number of devices, some of them having the same manufacturer, lowered success rates in all cases (6.56 % in the best case). However, it is considered that the decrease is not extremely pronounced, considering that there are 3 more devices and particularly 3 pairs of devices from the same manufacturer. It is important to note that the LG E400 device has in all cases the lowest success rates by far compared to the other devices (43.71 % average success rate). 31 %, 38 %, 38 % and 32 % of the images from the LG E400 (Optimus L3) were classified as images from the LG P760 (Optimus L9) in the first, second, third and fourth experiments in this set respectively. This clearly indicates a great level of confusion between the LG E400 and LG P760 images, which significantly lowers the overall success rate of each experiment. From the results it can be deduced that the technology and hardware and software components of both cameras could be similar (even if there are two intermediate models, the Optimus L5 and L7), or the defined feature set does not allow us to properly discern between the two cameras. It is also found, as in the previous experiment, that the Haar wavelet is not suitable for classifying images from mobile devices. For investigating deeper on the results of distinction between cameras from the same manufacturer the next set of experiments was made, in which we use 4 mobile device models of Samsung manufacturer. The results of the experiments are shown in Table 6.

Table 5 Image source acquisition identification for 10 smartphones

Features	Crop Size	Crop Align	I1	I2	HU	LG1	LG2	N1	BB	S1	S4	SE1	Average
All Features(Daubechies 8-tap)	1024x768	Center	91	87	96	47	89	92	95	61	79	80	79.98
All Features(Haar)	1024x768	Center	84	76	81	43	78	64	64	41	82	89	68.04
All Features(Daubechies 8-tap)	800x600	Center	91	85	95	43	80	88	97	52	72	82	76.23
All Features(Daubechies 8-tap)	640x480	Center	92	77	99	42	84	90	90	45	73	84	74.86

Table 6 Image source acquisition identification for 4 mobile devices of the same manufacturer

Features	Crop Size	Crop Align	S1	S2	S3	S4	Average
All Features (Daubechies 8-tap)	1024x768	Center	93	84	67	81	80.69
IQM + Color + Wavelet Daubechies 8-tap	1024x768	Center	88	84	50	84	74.65
hline All Features (Daubechies 8-tap)	800x600	Center	89	81	61	86	78.42
All Features (Daubechies 8-tap)	640x480	Center	88	78	63	77	75.96

Firstly we observe that the results using all features except noise are worse than those obtained when using all the features, as we could see in the experiments of Table 4. Having said this, it is observed that the results are quite good, because in the best case we obtain a 80.69 % average success rate. The results obtained for the Samsung GT-I8160P are worse compared with the results of the other devices, in particular a 59.90 % average success rate, when the average success rate in the worst case for the rest of the devices is 81.71 %. The 20 %, 33 %, 24 % and 23 % of the images of the Samsung GT-I8160P (Galaxy Ace 2) were classified as images of the Samsung GT-S5830M (Galaxy Ace) for the first, second, third and fourth experiment of this set, respectively. In fact, this concrete result, reduce a 5.16 % the best success rate obtained. Similarly to the previous case, can be supposed the same conclusions, although in this case the similarity of the cameras in all levels has more sense, since there are not intermediate models between the two devices in Ace Samsung series, that is, one succeeds the other.

5 Conclusions and discussion

Large numbers of digital images are circulating daily on the Internet or are used as evidence or proof in judicial proceedings. As a consequence, forensic analysis of digital images generated by devices as digital camera, mobile devices, scanner or computer becomes important in many real-life situations. It is noteworthy that forensics specific images techniques are required for mobile devices, not to be valid in most cases because there are significant intrinsic features which differentiate these types of devices. An example of this is presented in a situation where a forensic analyst needs to identify the type (camera, scanner, computer) or class (make and model) of the image acquisition source.

In this work we have presented various techniques for identifying mobile device images with respect to scanned and computer-generated images. Besides, other techniques that allow us to distinguish the acquisition source of smartphone images are presented. The techniques are based on the use of four feature sets (Noise, Color, IQM and Wavelets), on which adjustments have been made in order to improve the results for this specific type of devices. There have been experiments with the combination of the different feature sets, different crop sizes and positions, and wavelet functions.

With regard to source type identification, the first general conclusion is that Noise features are discarded as invalid when the number of types of devices is greater than 2. In the experiments that used whole images and different crop sizes and positions, unacceptable results were obtained for identifying three types of devices (scanner, smartphone and computer). As discussed in the experiments, for these three types of devices there are dozens of different manufacturers and models, hampering classification.

As a counterpart, forensic analysts may consider the application of the technique with Noise features for identifying the source type of images from mobile devices with respect to images from scanners and computers. The results are quite good at identifying the type when discerning between scanners and smartphones.

The use of all the features significantly improves results, but as a general conclusion they are not good enough to be used in a serious situation.

When identifying the acquisition source of mobile device images, the results are much more encouraging. In all sets of experiments performed, there is at least one configuration that yields good results, always putting them into the context of the level of demand on this technique (a large number of devices or many devices from the same manufacturer).

The use of all feature sets or wavelets with Daubechies 8-tap are the ones yielding better results. Regarding crop size, there is an optimal size for obtaining optimal results, which does not necessarily have to be the largest or the whole image, since the latter produces worse results than when using a crop. When taking a sufficiently large crop size, for example 1024x768, reducing the crop lowers the success rate. Regarding the number of devices used, as expected, the larger number of devices the lower the success rate. The same holds true when devices from the same manufacturer, whose cameras are similar or identical in some cases, are used.

The information from these conclusions can be a starting point for further works to optimize the success rate for the different cases presented with mobile device images. The choice of feature sets that offer better results and the remaining configuration parameters of the technique must be taken into account.

Therefore, the forensic analyst, knowing a priori information in some cases, and taking these conclusions into account, must decide on setting the various parameters of the technique and the validity of the results, taking into account the percentages obtained in the experiments presented in this work. That is to say, a single application of the technique can yield good results in some cases and bad results in others, depending on factors such as whether or not we want to decide the type or make and model of the devices, the number of devices used or the number of devices by the same manufacturer, amongst other things.

Acknowledgments Part of the computations of this work were performed in EOLO, the HPC of Climate Change of the International Campus of Excellence of Moncloa, funded by MECD and MICINN.

References

1. Ahonen T, Moore A (2012) Tomi Ahonen Almanac 2012: mobile telecoms industry annual review
2. Al-Zarouni M (2006) Mobile handset forensic evidence: a challenge for law enforcement. In: Proceedings of the 4th australian digital forensics conference. school of computer and information science, edith cowan university
3. Baer R (2010) Resolution limits in digital photography: the looming end of the pixel wars - OSA technical digest (CD). In: Proceedings of the imaging systems, p. ITuB3. Optical society of America. doi:10.1364/IS.2010.ITuB3
4. Bayram S, Sencar HT, Memon N (2006) Improvements on source camera-model identification based on CFA interpolation. In: Working group 11.9 international conference on digital forensics, pp. 24–27. Springer
5. Bayram S, Sencar HT, Memon N (2008) Classification of digital camera-models based on demosaicing artifacts. Digit Investig 5(1-2):49–59. doi:10.1016/j.diin.2008.06.004
6. Boutell M, Luo J (2004) Photo classification by integrating image content and camera metadata. In: Proceedings of the 17th international conference on pattern recognition, vol. 4, pp. 901–904. IEEE Computer Society. doi:10.1109/ICPR.2004.1333918

7. Boutell M, Luo J (2005) Beyond pixels: exploiting camera metadata for photo classification. Pattern Recognit 38(6):935–946. doi:10.1016/j.patcog.2004.11.013
8. Cao H, Kot AC (2010) Mobile camera identification using demosaicing features. In: Circuits and systems (ISCAS), Proceedings of 2010 IEEE international symposium on, pp. 1683–1686. IEEE
9. Celiktutan O, Avcibas I, Sankur B, Ayerden NP, Capar C (2006) Source cell-phone identification. In: Proceedings of the IEEE 14th signal processing and communications applications, pp. 1–3. IEEE. doi:10.1109/SIU.2006.1659882
10. Celiktutan O, Sankur B, Avcibas I (2008) Blind identification of source cell-phone model. IEEE Trans Inf Forensics Secur 3(3):553–566
11. Chang CC, Lin CJ LIBSVM: a library for support vector machines. Version 3.17, Abril 26, 2013 http://www.csie.ntu.edu.tw/cjlin/libsvm/
12. Chen M, Fridrich J, Goljan M, Lukas J (2008) Determining image origin and integrity using sensor noise. IEEE Trans Inf Forensics Secur 3(1):74–90. doi:10.1109/TIFS.2007.916285
13. Choi KS (2006) Source camera identification using footprints from lens aberration. In: Proceedings on digital photography II, no. 852 in 6069, pp. 60,690J–60,690J–8. SPIE international society for optical engineering
14. Committee S Exchangeable Image File for digital still cameras: Exif version 2.3, April 26, 2010. http://www.cipa.jp/english/hyoujunka/kikaku/pdf/DC-008-2010_E.pdf
15. Costa FDO, Eckmann M, Scheirer WJ, Rocha A (2012) Open set source camera attribution. In: Proceedings of the 25th conference on graphics, patterns and images, pp. 71–78. IEEE. doi:10.1109/SIBGRAPI.2012.19
16. Fan J, Kot A, Cao H, Sattar F (2011) Modeling the EXIF-image correlation for image manipulation detection. In: 18th IEEE international conference on image processing (ICIP), pp. 1945–1948. doi:10.1109/ICIP.2011.6115853
17. Gartner Inc. (2013) Gartner says smartphone sales grew 46.5 percent in second quarter of 2013 and exceeded feature phone sales for First Time http://www.gartner.com/newsroom/id/2665715
18. Geradts ZJ, Bijhold J, Kieft M, Kurosawa K, Kuroki K, Saitoh N (2001) Methods for identification of images acquired with digital cameras. In: Proceedings on enabling technologies for law enforcement and security, vol. 4232, pp. 505–512. spie-international society for optical engine. doi:10.1117/12.417569
19. Gloe T, Kirchner M, Winkler A, Bohme R (2007) Can we trust digital image forensics? In: Proceedings of the 15th international conference on multimedia, pp. 78–86. ACM Press. doi:10.1145/1291233.1291252
20. Ho JS, Au OC, Zhou J, Guo Y (2010) Inter-channel demosaicking traces for digital image forensics. In: Multimedia and expo (ICME), 2010 IEEE international conference on, pp. 1475–1480. doi:10.1109/ICME.2010.5582951
21. Hsuand CW, Chang CC, Lin CJ (2003) A Practical Guide to Support Vector Classification. Practical guide, Department of Computer Science and Information Engineering, National Taiwan University
22. Hu Y, Li CT, Zhou C (2010) Selecting Forensic Features for Robust Source Camera Identification. In: Computer Symposium (ICS), 2010 International, pp. 506–511. doi:10.1109/COMPSYM.2010.5685458
23. Embedded Imaging Takes Off as Stand-alone Digital Cameras Stall (2013). http://www.icinsights.com/data/articles/documents/484.pdf
24. Jang CJ, Lee JY, won Lee J, Cho HG (2007) Smart management system for digital photographs using temporal and spatial features with EXIF metadata. In: Digital information management, 2007. ICDIM '07. 2nd international conference on, vol. 1, pp. 110–115. doi:10.1109/ICDIM.2007.4444209
25. Khanna N, Mikkilineni AK, Delp EJ (2009) Scanner identification using feature-based processing and analysis. IEEE Trans Inf Forensics Secur 4(1):123–139
26. Khannaa N, Mikkilinenib AK, Chiub GT, Allebacha J, Delpa EJ (2006) Forensic Classification of Imaging Sensor Types. Rfc, Purdue University
27. Liu Q, Li X, Chen L, Cho H, Cooper AP, Chen Z, Qiao M, Sung AH (2012) Identification of smartphone-image source and manipulation. In: Jiang H, Ding W, Ali M, Wu X (eds) Advanced research in applied artificial intelligence, lecture notes in computer science, vol. 7345. Springer Berlin Heidelberg, Dalian, pp 262–271. doi:10.1007/978-3-642-31087-4_28
28. Long Y, Huang Y (2006) Image based source camera identification using demosaicking. In: Proceedings of the IEEE 8th Workshop on multimedia signal processing, pp. 419–424. IEEE. doi:10.1109/MMSP.2006.285343
29. Lukas J, Fridrich J, Goljan M (2006) Digital camera identification from sensor pattern noise. IEEE Trans Inf Forensics Secur 1(2):205–214. doi:10.1109/TIFS.2006.873602
30. McKay C (2007) Forensic analysis of digital imaging devices. Technical report, University of Maryland

31. Mckay C, Swaminathan A, Gou H, Wu M (2008) Image acquisition forensics: forensic analysis to identify imaging source. In: Proceedings of the IEEE international conference on acoustics, speech and signal processing, international conference on acoustics speech and signal processing (ICASSP), pp. 1657–1660. IEEE. doi:10.1109/ICASSP.2008.4517945
32. Michie D, Spiegelhalter DJ, Taylor CC (1994) Machine learning, neural and statistical classification. Ellis Horwood
33. de Costa FO, Silva E, Eckmann M, Scheirer WJ, Rocha A (2014) Open set source camera attribution and device linking. Pattern Recogn Lett 39(0):92–101
34. Ozparlak L, Avcibas I (2011) Differentiating between images using wavelet-based transforms: a comparative study. IEEE Trans Inf Forensics Secur 6(4):1418–1431
35. Platt J (2000) AutoAlbum: Clustering Digital Photographs Using Probabilistic Model Merging. In: Proceedings of the IEEE Workshop on Content-based Access of Image and Video Libraries, pp. 96–100. IEEE. doi:10.1109/IVL.2000.853847
36. Rocha A, Scheirer W, Boult T, Goldenstein S (2011) Vision of the unseen: current trends and challenges in digital image and video forensics. ACM Comput Surv 43(4):26:1–26:42. doi:10.1145/1978802.1978805
37. Romero NL, Chornet VG, Cobos JS, Carot AS, Centellas FC, Mendez MC (2008) Recovery of descriptive information in images from digital libraries by means of EXIF metadata. Library Hi Tech 26(2):302–315. doi:10.1108/07378830810880388
38. Rosales Corripio J, Arenas González DM, Sandoval Orozco AL, García Villalba LJ, Hernandez-Castro JC, Gibson SJ (2013) Source smartphone identification using sensor pattern noise and wavelet transform. In: Proceedings of the 5th international conference on imaging for crime detection and prevention (ICDP 2013), pp. 1–6
39. Sandoval Orozco A, Arenas González D, Rosales Corripio J, García Villalba L, Hernandez-Castro J (2013) Techniques for source camera identification. In: Proceedings of the 6th international conference on information technology, pp. 1–9
40. Sandoval Orozco AL, Arenas González DM, García Villalba LJ, Hernandez-Castro J (2014) Analysis of Errors in Exif Metadata on Mobile Devices. Multimedia Tools and Applications pp. 1–29. doi:10.1007/s11042-013-1837-6
41. Sandoval Orozco AL, Arenas Gonzlez DM, Rosales Corripio J, Garca Villalba LJ, Hernandez-Castro JC (2014) Source identification for mobile devices, based on wavelet transforms combined with sensor imperfections. Computing 96(9):829–841. doi:10.1007/s00607-013-0313-5
42. Swaminathan A, Wu M, Liu K (2009) Component forensics. IEEE Signal Process Mag 26(2):38–48. doi:10.1109/MSP.2008.931076
43. Tesic J (2005) Metadata practices for consumer photos. IEEE Multimedia 12(3):86–92. doi:10.1109/MMUL.2005.50
44. Tsai MJ, Lai CL, Liu J (2007) Camera/Mobile Phone Source Identification for Digital Forensics. In: Proceedings of the International Conference on Acoustics Speech and Signal Processing, pp. II–221–II–224. IEEE. doi:10.1109/ICASSP.2007.366212
45. Van LT, Emmanuel S, Kankanhalli M (2007) Identifying Source Cell Phone using Chromatic Aberration. In: IEEE International Conference on Multimedia and Expo, pp. 883–886. doi:10.1109/ICME.2007.4284792
46. Van Lanh T, Chong KS, Emmanuel S, Kankanhalli MS (2007) A Survey on Digital Camera Image Forensic Methods. In: IEEE International Conference on Multimedia and Expo, pp. 16–19. IEEE. doi:10.1109/ICME.2007.4284575
47. Wang B, Guo Y, Kong X, Meng F (2009) Source Camera Identification Forensics Based on Wavelet Features. In: International Conference on Intelligent Information Hiding and Multimedia Signal Processing, vol. 0, pp. 702–705. IEEE Computer Society. doi:10.1109/IIH-MSP.2009.244

Ana Lucila Sandoval Orozco received a Computer Science Engineering degree from the Universidad Autónoma del Caribe (Colombia) in 2001. She holds a Specialization Course in Computer Networks (2006) from the Universidad del Norte (Colombia) and holds a M.Sc. in Research in Computer Science from the Universidad Complutense de Madrid (Spain) in 2009. She is currently a Ph.D. Student at the Universidad Complutense de Madrid (Spain) and a Research Assistant at Complutense Research Group GASS. Her main research interests are coding theory, information security and its applications.

Jocelin Rosales Corripio received a Computer Science Engineering degree from the Benemerita Universidad Autónoma de Puebla (Mexico) in 2008. She holds a M.Sc. in Research in Computer Science from the Universidad Complutense de Madrid (Spain) in 2013. She is currently a Ph.D. Student at the Universidad Complutense de Madrid (Spain) and a Research Assistant at Complutense Research Group GASS. Her main research interests are coding theory, information security and its applications.

Luis Javier García Villalba received a Telecommunication Engineering degree from the Universidad de Málaga (Spain) in 1993 and holds a M.Sc. in Computer Networks (1996) and a Ph.D. in Computer Science (1999), both from the Universidad Politcnica de Madrid (Spain). Visiting Scholar at COSIC (Computer Security and Industrial Cryptography, Department of Electrical Engineering, Faculty of Engineering, Katholieke Universiteit Leuven, Belgium) in 2000 and Visiting Scientist at IBM Research Division (IBM Almaden Research Center, San Jose, CA, USA) in 2001 and 2002, he is currently Associate Professor of the Department of Software Engineering and Artificial Intelligence at the Universidad Complutense de Madrid (UCM) and Head of Complutense Research Group GASS (Group of Analysis, Security and Systems) which is located in the School of Computer Science at the UCM Campus. His professional experience includes research projects with Hitachi, IBM, Nokia and Safelayer Secure Communications. His main research interests are cryptography, coding, information security and its applications.

Julio César Hernández Castro received the B.Sc. degree in Mathematics from the Universidad Complutense de Madrid, Madrid, Spain, in 1995, the M.Sc. degree in Coding Theory and Network Security from the Universidad de Valladolid in 1999, and the PhD degree in Computer Science from Carlos III University of Madrid in 2003. He is currently a Lecturer in Computer Security at the School of Computing, University of Kent. He is also associated with Kent Cyber Security Center. His interests include Cryptology, Steganography & Steganalysis, Computer & Network Security, Computer Forensics, CAPTCHAs, RFID Security, and the application of NonStandard techniques to Cryptology.

Pep Lluís Ferrer Gomila · M. Francisca Hinarejos Campos
(editores)

Actas de la XIV Reunión Española sobre Criptología y Seguridad de la Información

Actas de la XIV Reunión Española sobre Criptología y Seguridad de la Información

RECSI XIV

Maó, Menorca, Illes Balears, 26-28 Octubre de 2016

Publicado por:
Departamento de Ciencias Matemáticas e Informática
Universidad de les Illes Balears
Ctra. de Valldemossa, km 7.5. Palma (Illes Balears)
http://recsi16.uib.es

ISBN: 978-84-608-9470-4

Créditos:

Primera edición – Octubre 2016

Organizadores

Colaboradores

Theia: Una Herramienta para el Análisis Forense de Imágenes Digitales de Dispositivos Móviles

Jocelin Rosales Corripio, Anissa El-Khattabi, Ana Lucila Sandoval Orozco, Luis Javier García Villalba
Grupo de Análisis, Seguridad y Sistemas (GASS)
Departamento de Ingeniería del Software e Inteligencia Artificial
Facultad de Informática, Despacho 431, Universidad Complutense de Madrid (UCM)
Calle Profesor José García Santesmases, 9, Ciudad Universitaria, 28040 Madrid
Email: jocelinr@ucm.es, {asandoval, javiergv}@fdi.ucm.es

***Resumen*—El número de cámaras digitales integradas en dispositivos móviles así como su uso en la vida cotidiana está en continuo crecimiento. Diariamente gran cantidad de imágenes generadas por este tipo de dispositivos circulan en Internet o son utilizadas como evidencias o pruebas en procesos judiciales. Como consecuencia, el análisis forense de imágenes digitales de dispositivos móviles cobra importancia en multitud de situaciones de la vida real. Cabe destacar que se necesitan técnicas forenses específicas para imágenes de dispositivos móviles, no siendo válidas en muchos casos las técnicas utilizadas para las cámaras digitales tradicionales, debido a las características intrínsecas que diferencian ambos tipos de cámaras. Este trabajo presenta una herramienta para el análisis forense de imágenes digitales de dispositivos móviles denominada *Theia*. Entre las diversas ramas del análisis forense, *Theia* se centra principalmente en la identificación fiable de la marca y modelo de la cámara móvil que generó una imagen dada.**

***Palabras clave*—Análisis forense de imágenes digitales, identificación de la fuente de adquisición, (Digital image forensics analysis, image source adquisition identification)**

I. Introducción

Según estimaciones de la Unión Internacional de Telecomunicaciones (UIT), en 2014 hubo 6,87 miles de millones de líneas dadas de alta en dispositivos móviles en todo el mundo, lo que supone un incremento sobre los 6,62 miles de millones de suscripciones de 2013 y los 6,19 miles de millones de 2012. Asimismo, la UIT estima que en 2014 hubo 2,29 miles de millones de altas de líneas de banda ancha en dispositivos móviles, suponiendo un notable aumento sobre los 1,91 y 1,54 miles de millones de altas de los años 2013 y 2012, respectivamente. La irrupción de este tipo de dispositivos en el día a día es tal que un gran número de personas tienen y usan más de un dispositivo móvil y un usuario típico en promedio consulta su móvil unas 150 veces al día, siendo 8 de ellas para hacer uso de la funcionalidad de la cámara.

En los países industrializados el 97 % de teléfonos móviles incorpora una cámara digital integrada. Asimismo, la mayor parte del resto de tipos de dispositivos móviles también posee una cámara digital integrada. Estas cámaras, a diferencia de las cámaras digitales tradicionales (DSC), son llevadas por sus dueños gran parte del tiempo a la mayoría de los lugares a los que asisten [1]. En el año 2016 la venta de DSCs descenderá de un 47 % de cuota de mercado sobre el total de cámaras digitales que obtuvo en el 2012 a un 27 %. Asimismo, se prevé un incremento en las ventas de cámaras digitales integradas en teléfonos móviles, PC y tabletas, de un 31 % de cuota de mercado sobre el total de cámaras digitales en 2012, a un 42 % en el 2016 [2].

Prácticamente la totalidad de estas cámaras digitales tienen funciones de grabación de vídeo. Actualmente, existe una gran competencia entre los fabricantes por integrar una videocámara de alta definición al alcance del usuario en todo momento. Como consecuencia, y dada la gran cantidad de tiempo que una persona pasa junto a los dispositivos móviles, éstos se han convertido para muchas personas en el primer dispositivo de captura de fotografías y grabación de vídeos. El amplio uso de cámaras digitales en dispositivos móviles es una realidad en la vida cotidiana. Diariamente pueden verse imágenes generadas por dispositivos móviles en telenoticias, correo electrónico o en redes sociales. Webs como *Facebook*, *Youtube* o *Twitter* entre otras, se sitúan en los puestos más altos de la lista de webs más visitadas, siendo una parte considerable de su contenido capturado con cámaras digitales de dispositivos móviles [3]. Todo esto hace que en ciertos casos existan restricciones legales o limitaciones a su utilización en lugares como: colegios, universidades, oficinas de gobierno, empresas, etc. Además, y como consecuencia de todo lo anterior, cada día las imágenes digitales generadas con dispositivos móviles son más utilizadas como testigos silenciosos en procesos judiciales (pornografía infantil, espionaje industrial, violencia callejera, redes sociales, ...), siendo en muchos casos piezas cruciales de la evidencia de un crimen [4, 5].

Por tanto, el análisis forense de imágenes digitales de dispositivos móviles cobra mucha fuerza actualmente. El estudio debe ser concreto para este tipo de dispositivos, ya que poseen características específicas que permiten obtener mejores resultados, no siendo válidas las técnicas forenses para imágenes digitales generadas por otros tipos de dispositivos.

El resto del trabajo está estructurado como sigue: La sección II resume las principales técnicas de análisis forense para la identificación de la fuente de adquisición de imágenes digitales. La sección III especifica *Theia*, una herramienta para el análisis forense de imágenes digitales. La sección IV presenta distintos análisis realizados a bancos de imágenes con *Theia*. Por último, las principales conclusiones de este trabajo se presentan en la sección V.

II. Análisis Forense de Imágenes Digitales

Como los dispositivos móviles proliferan a un ritmo imparable en nuestra sociedad y los avances en las tecnologías de semiconductores permiten que éstos amplíen su capacidad de procesamiento y almacenamiento, estos dispositivos son frecuentemente evidencias en procesos judiciales o investigaciones policiales de todo tipo. A continuación se presentan algunas situaciones donde se aprecia la necesidad de herramientas para el análisis forense en estos dispositivos [6].

Un escenario de interés es que los dispositivos móviles son centro de envíos de mensajes (SMS, MMS, redes sociales, aplicaciones específicas) y correos electrónicos. Por tanto, hoy en día, los dispositivos móviles son fuente de adquisición, tratamiento y almacenamiento de información relevante de distinta naturaleza. Otro escenario, es la utilización de los dispositivos móviles como centros de transacciones en línea: transacciones bancarias, compras en Internet, reservas de vuelos y hoteles, etc., donde se realizan operaciones con datos sensibles. En el caso concreto del análisis forense de imágenes de dispositivos móviles no hay duda de la importancia que puede tener su aplicación en casos judiciales. Dado el gran uso de este tipo de cámaras y las polémicas que suscitan, son muchos los debates y normas que prohíben su uso. Asimismo, los dispositivos móviles pueden albergar fotografías almacenadas de carácter personal o que hayan sido tomadas in situ. Estas fotografías pueden ser evidencias de un hecho y elementos potenciales de uso en procesos judiciales y, consecuentemente, elementos de estudio del análisis forense.

El análisis forense de imágenes digitales se divide en dos grandes ramas [7]: autenticidad de imágenes digitales e identificación de la fuente de adquisición de una imagen. La primera trata de discernir si una imagen ha sufrido algún procesamiento posterior al de su creación, es decir, que no haya sido manipulada. La segunda pretende identificar la marca y modelo del dispositivo que generó la imagen digital. La técnica más utilizada para la identificación de la fuente es la máquina de soporte vectorial (SVM) [8].

El éxito de las técnicas de identificación de la fuente de la imagen depende del supuesto de que todas las imágenes adquiridas por un mismo dispositivo presentan características intrínsecas del mismo. Las características que se usan para identificar marca y modelo de las cámaras digitales se derivan de las diferencias que existen entre las técnicas de procesamiento de las imágenes y las tecnologías de los componentes que se utilizan. Para el diseño de técnicas y algoritmos en cualquiera de estas ramas se aprovechan algunas características especiales de las imágenes que sirven como herramienta para el análisis forense. El mayor problema con este enfoque es que los diferentes modelos de las cámaras digitales usan componentes de un número reducido de fabricantes y que los algoritmos que usan para la generación de las imágenes también son muy similares entre modelos de la misma marca.

Según [9] se pueden establecer cuatro grupos de técnicas para este fin: basadas en la aberración de las lentes, basadas en la interpolación de la matriz CFA, basadas en el uso de las características de la imagen y basadas en las imperfecciones del sensor. Asimismo, existe otro grupo de técnicas forenses a destacar: las basadas en los metadatos de la imagen.

Las técnicas basadas en metadatos dependen en gran medida de los metadatos que los fabricantes deciden insertar cuando la imagen es generada. Estos metadatos embebidos en los archivos de las imágenes digitales son una poderosa fuente de información. Los metadatos o "datos sobre datos" registran información relacionada con las condiciones de captura de la imagen, como fecha y hora de generación, presencia o ausencia de *flash*, distancia de los objetos, tiempo de exposición, apertura del obturador e información GPS, entre otras. En otras palabras, información de interés que complementa el contenido principal de un documento digital. Los metadatos, entre otros usos, pueden llegar a ser una potente ayuda para la organización y búsqueda en librerías de imágenes. La especificación Exif [10] es la más utilizada para identificación de la fuente por ser el contenedor de metadatos más común en las cámaras digitales [11]. La especificación Exif incluye cientos de etiquetas, entre las que se encuentran *marca* y *modelo*, aunque la propia especificación no hace obligatoria su existencia en los archivos. [12, 13] realizan un estudio a fondo, donde se demuestra que los fabricantes no siguen fielmente la especificación Exif. Esto puede conllevar la extracción de información errónea o inválida para fines forenses. Asimismo, este método es el más vulnerable a modificaciones malintencionadas, e incluso se puede dar el caso de la eliminación total de los metadatos, ya sea intencionadamente o de manera inconsciente. Ejemplos de ello son algunos programas de edición fotográfica, que al editar o comprimir una imagen, actualizan incorrectamente los metadatos o provocan la pérdida de los mismos. A pesar de las debilidades de este tipo de técnicas, si existen en el archivo los metadatos y de alguna manera se logra comprobar que no han sufrido modificaciones externas, su uso es de gran utilidad para los analistas forenses. Existe información difícilmente inferible del propio contenido de la imagen como por ejemplo la información GPS o la fecha y hora de la toma de la imagen, entre muchas otras.

Las técnicas basadas en las características de las imágenes utilizan un conjunto de características extraídas del contenido de la escena de la imagen para realizar la identificación de la fuente. Existen tres grandes grupos de características clasificadas por su tipología: características de color, métricas de calidad de la imagen (IQM) y estadísticas del dominio wavelet [14–18] .

Las técnicas basadas en las imperfecciones del sensor estudian las huellas que los defectos del sensor pueden dejar sobre las imágenes. Estas técnicas se dividen en dos ramas: defectos de píxel y patrón del ruido del sensor (SPN). En la primera se estudian los defectos de píxel, píxeles calientes, píxeles muertos, defectos de fila o columna y los defectos de grupo. En la segunda se construye un patrón del ruido promediando los múltiples residuos de ruido obtenidos mediante algún filtro de eliminación de ruido. La presencia del patrón se determina utilizando un método de clasificación (correlación o máquinas SVM) [9, 19].

III. Theia: Herramienta para el Análisis Forense de Imágenes de Dispositivos Móviles

En este trabajo se ha desarrollado una herramienta para el análisis forense de imágenes digitales de dispositivos móviles denominada *Theia*. Su objetivo versa principalmente sobre el diseño de diversas técnicas para la identificación de la fuente de adquisición de imágenes digitales generadas con dispositivos móviles.

Theia permite la gestión avanzada de proyectos independientes con distintos conjuntos de imágenes y tiene distintas funcionalidades a nivel de tratamiento de imágenes, tanto individualmente como en grupo. El tratamiento grupal de imágenes permite al analista forense realizar de forma rápida, sencilla y con gran versatilidad diversos análisis sobre grandes bancos de imágenes. Las principales funcionalidades de *Theia* son: (i) Tratamiento de metadatos Exif para la identificación de la fuente de adquisición; (ii) Detección de manipulaciones basadas en el procesado de la imagen en miniatura; (iii) Identificación de la fuente de adquisición de imágenes basado en las características de la imagen y (iv) Clasificación sin supervisión de Imágenes digitales.

III-A. Tratamiento de Metadatos Exif para la Identificación de la Fuente de Adquisición

Theia sirve de apoyo a la tarea del analista forense en lo que respecta al análisis de los metadatos, haciendo menos complejo realizar el procesamiento para un gran número de imágenes. No es simplemente un tema de ayuda al analista forense con respecto al rendimiento en el procesamiento de un gran número de imágenes, sino que aporta fiabilidad al proceso y ofrece diferentes funcionalidades complementarias. El análisis de metadatos se divide en dos grandes partes:

III-A1. Tratamiento de imágenes a nivel individual: Esta funcionalidad está asociada a la pestaña Exif Info. Permite obtener la información Exif detallada de una imagen individual, encontrar modificaciones realizadas en la imagen al compararla con la imagen en miniatura existente en la información Exif y situar la imagen en Google Maps y Google Earth (si posee información de geoposicionamiento) como se puede observar en la Figura 1. A la hora de mostrar la información Exif se ha organizado en 6 grupos: *Image*, *Exif*, *GPS*, *Interoperability*, *Thumbnail* y *Maker Note*.

Image Info: Almacena la información relativa a la propia imagen y que no tienen relación directa con el entorno y el momento de la captura.

Exif Info: Guarda las etiquetas con información relativa al momento o al entorno de la toma de la imagen.

GPS Info: Se encuentra la información relativa al geoposicionamiento.

InterOperability Info: Se incluyen las etiquetas relativas a la información de las reglas de interoperabilidad, como pueden ser Exif R98, DCF thumbnail file o DCF Option file.

Thumbnail Info: Se almacenan las etiquetas relativas a la información de la imagen en miniatura.

Maker Note Info: Almacena la información que cada fabricante puede insertar de forma opcional y que no ha sido recogida en ninguna etiqueta Exif. Esta información tiene un formato libre y no tiene una estructura prefijada.

III-A2. Tratamiento de imágenes a nivel de grupo: Esta funcionalidad está asociada a la pestaña DDBB Projects (Figura 2). Permite hacer análisis de imágenes en grupo. Cada grupo es totalmente independiente entre sí. Su estructura es mucho más compleja que la de la pestaña Exif Info. Asimismo, ofrece gran diversidad de opciones al analista forense: Gestión de imágenes en proyectos, consultas preestablecidas, consultas avanzadas y geoposicionamiento de las imágenes.

Gestión de imágenes en proyectos: Las imágenes se procesan en grupos denominados proyectos. Cada proyecto es totalmente independiente entre sí. Se busca acercar la realidad del día a día del analista forense a la herramienta, es decir, el analista tendrá diversos casos de análisis disjuntos, los cuales podrá tratar en proyectos distintos. Para cada imagen se muestra información básica obtenida de los metadatos Exif. Además se presenta la información de si posee metadatos en los distintos grupos Exif que analiza la herramienta. Asimismo, se muestra la imagen seleccionada, la imagen en miniatura original almacenada en el archivo de la imagen (si la posee) y la imagen en miniatura generada por *Theia*.

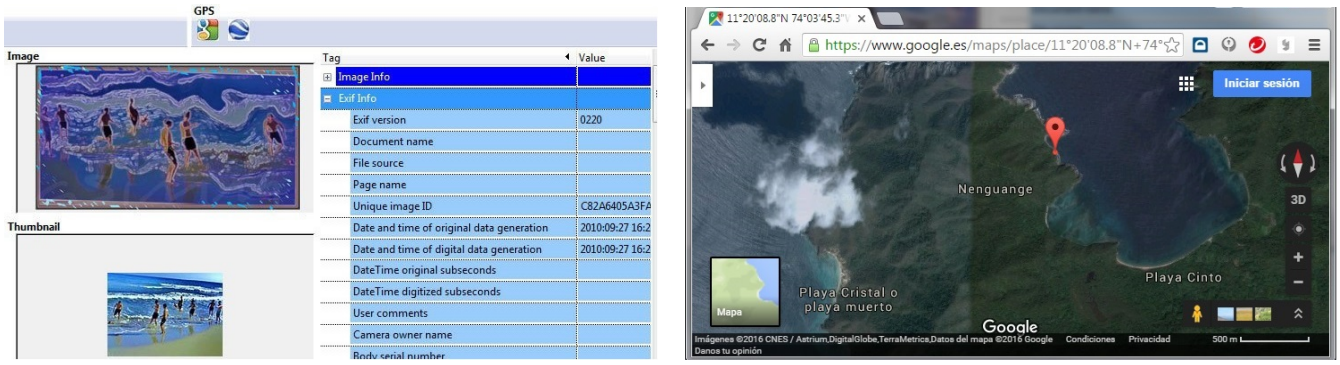
Consultas preestablecidas: Permite crear consultas agregando etiquetas Exif sobre las imágenes del grupo seleccionado. La consulta agrupa las imágenes por los criterios seleccionados y muestra el número de imágenes que hay en cada uno de los grupos formados.

Consultas avanzadas: Permite la creación de consultas sobre imágenes de un grupo, configurando los datos Exif a mostrar y los filtros a aplicar. Es decir, muestra la información de las imágenes de los campos seleccionados que coincidan con uno de los valores de cada uno de los filtros configurados. Asimismo, se permite el almacenamiento permanente de consultas.

Geoposicionamiento: Permite el tratamiento de la información de geoposicionamiento de las imágenes de un proyecto que posean esta información. Se genera un mapa utilizando la API de Google Maps que situa a las imágenes que información de geoposicionamiento. En el mapa se agrupan las imágenes por zona y, a medida que se aumenta el *zoom*, se van detallando las coordenadas. La Figura 3 muestra un ejemplo del mapa generado, *Theia* permite aumentar el *zoom* en una zona concreta.

III-B. Detección de Manipulaciones basado en el Procesado de la Imagen Miniatura

Tiene como objetivo determinar si se realizaron modificaciones en las imágenes de un proyecto posteriores a la captura de las mismas. Esta operación se realiza calculando el valor cuadrático medio o RMS (del inglés *root mean square*) de la comparación de la imagen en miniatura que se encuentra en la información Exif con la imagen en miniatura generada port *Theia* a partir de la imagen analizada. El resultado de esta comparación se representa con color entre tres posibles: verde (no modificadas), amarillo (posiblemente modificadas) o rojo (modificadas) (ver Figura 4).



(a) Análisis individual de una imagen

(b) Geoposicionamiento de una imagen en Google Maps

Figura 1. Apariencia general del tratamiento de imágenes a nivel individual

Figura 2. Apariencia general de la pestaña DDBB Projects

Figura 4. Análisis de la imagen en miniatura

Figura 3. Geoposicionamiento de un grupo de imágenes en Google Maps

III-C. Identificación de la fuente de adquisición de imágenes basado en las características de la imagen

Theia permite la identificación de la fuente de adquisición de un conjunto de imágenes de un proyecto. Las técnicas utilizadas se basan en las características del contenido de la imagen. Las funciones son las siguientes: extracción de características, entrenamiento y clasificación con la máquina SVM.

Extracción de características: Para iniciar el proceso de identificación de la fuente de adquisición es requisito extraer las características de las imágenes involucradas. Hay dos formas de extracción de características de las imágenes de un proyecto: en el momento de su creación o mediante la ventana de extracción de características. *Theia* permite seleccionar al usuario los conjuntos de características a extraer. Estos conjuntos de características son:

- "*Color Features*": La configuración de los filtros de la matriz CFA, el algoritmo de *demosaicing* y la técnicas aplicadas al procesamiento del color hacen que las señales contenidas en la bandas de color tengan tratamientos y patrones específicos.
- "*IQM Features*": Los diferentes modelos de cámara producen imágenes de diferente calidad. Puede haber diferencias en la luminosidad de la imagen, la nitidez o en la calidad del color. Estas diferencias hacen que se proponga un conjunto de métricas de calidad, como características que permiten diferenciar la fuente de las imágenes.
- "*Wavelets Features*": Debido a la propiedad determinista del patrón de ruido del sensor presente en una imagen, este patrón puede usarse como huella para identificar el dispositivo que generó la imagen.
- "*Noise Features*": El proceso de generación de imágenes suele introducir varios defectos en estas, los cuáles crearán ruido que aparecerá en la imagen final.

Las características extraídas se almacenan en la base de datos y quedan disponibles para su uso en distintas ejecuciones

de *Theia*, ya que los procesos de extracción de características de un gran número de imágenes pueden ser costosos en tiempo. Posteriormente, es necesaria la fase de entrenamiento del clasificador SVM. La ventana de configuración de la fase de entrenamiento se muestra en la Figura 5.

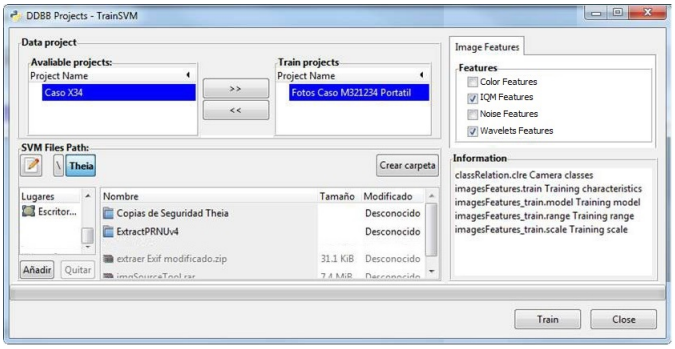


Figura 5. Configuración de la fase de entrenamiento de la máquina SVM

Una vez realizada la fase de entrenamiento se puede comenzar con la fase de clasificación o predicción, la cual es la que finalmente identifica la fuente de adquisición de las imágenes utilizando las características seleccionadas por el usuario.

En la fase de clasificación el usuario solo puede utilizar las imágenes de un proyecto para que se identifique su fuente de adquisición. El proyecto seleccionado será el utilizado para la fase de clasificación. Una vez terminado el proceso de clasificación de la fuente de adquisición, *Theia* almacena los resultados de la identificación de la fuente de adquisición de las imágenes del proyecto seleccionado.

III-D. Clasificación sin Supervisión de Imágenes Digitales

Numerosas situaciones de aplicación real de los algoritmos de identificación de la fuente de adquisición de imágenes digitales pueden darse en los denominados “escenarios abiertos”. Es decir, el analista forense no tiene conocimiento a priori del conjunto de dispositivos al que pertenecen las imágenes a clasificar. En estos casos el objetivo no es la identificación de la marca y modelo de las imágenes, sino la agrupación de las imágenes en clases o *clusters* por modelo de dispositivo fuente. En *Theia* se utiliza el algoritmo de *clustering* basado en el uso de las características del patrón del ruido del sensor diseñado en [20].

El objetivo del análisis de *clusters* o *clustering* de imágenes digitales es agrupar una colección de imágenes en conjuntos cuyas imágenes pertenecen a un mismo dispositivo llamadas *clusters* sin información a priori. La agrupación de imágenes puede llevarse a cabo mediante técnicas de aprendizaje supervisadas o sin supervisión. En el primer caso es indispensable conocer información del dispositivo a priori, es decir se identifica claramente con la clasificación en escenarios cerrados en donde se requiere una fase de entrenamiento con las características extraídas de las imágenes y una segunda fase de clasificación conforme al resultado anterior. Sin embargo, en un caso real puede ser difícil contar con la cámara en cuestión o con un conjunto de fotografías tomadas por la misma para llevar a cabo un entrenamiento, de ahí la necesidad de técnicas de aprendizaje sin supervisión, que se corresponden directamente con los escenarios abiertos. El algoritmo de agrupación sin supervisión es una combinación entre un *clustering* jerárquico y un *clustering* plano. Es decir, a pesar de formar una estructura de dendrograma con cada iteración del algoritmo, al final los *clusters* son tomados como entidades sin relación alguna ya que cada uno de ellos debe corresponder a un dispositivo específico.

IV. Análisis Forense con Theia

Para evaluar algunas de las funcionalidades de *Theia* se realizaron diferentes análisis sobre un conjunto de imágenes reales de dispositivos móviles. El objetivo del análisis es la búsqueda de datos de interés, patrones de valores o simplemente información estadística sobre los metadatos Exif del banco de imágenes. Las imágenes han sido obtenidas de dispositivos móoviles de personas conocidas intentando buscar la máxima heterogeneidad posible con respecto a las marcas y los modelos de los dispositivos, así como contar con el mayor número de imágenes de cada uno de ellos. El banco de imágenes está formado por 3751 imágenes de 10 marcas y 91 modelos. En la Tabla I se muestran algunos de los modelos analizados agrupados por marca con su correspondiente número de imágenes. A diferencia de los estudios realizados en otros trabajos relacionados, el número de modelos de cámaras utilizado es mucho mayor.

Tabla I
Dispositivos móviles clasificados por marca y modelo

Marca	Modelo
Apple	Ipad 2 (19), iPhone 3g (51) iPhone 3GS (82), iPhone 4g (49)
HTC	Desire HD (162), Droid incredible (32), Hero (59), TyTN ii (59)
LG	CU720 (31), Ku990i (144), Rumor (26), VX-8550 (13), VX9700 (30)
Nokia	5230 (19), 5300 (100), 6020 (30), E71 (5), N70 (16), N95 (131)
Samsung	Galaxy 3 (33), II (30), Omnia (37), SGH-F250L (4), Wave (17)
Sony Ericsson	C702 (79), C905 (40), T707 (102), W910i (7), Z610i (61)
Motorola	Atrix (35), Cliq (30), Droid (31), Droid x2 (54), W377 (20)

Como se ha cometado anteriormente, la imagen miniatura es una copia de la imagen original pero con un tamaño reducido. Tener la imagen miniatura almacenada es el primer paso para poder analizar si la imagen ha sido manipulada posteriomente a la captura. En los siguientes análisis se estudian las etiquetas Exif que se encuentran en el bloque “Thumbnail Info” con “Query Set”, examinando cuáles son las imágenes que no poseen información en este bloque. El resultado de este análisis, que se muestra en la Tabla II, revela que 2879 imágenes (el 76,75 %) posee información de la imagen en miniatura, frente a 872 que no la posee; de estas últimas, 456 son del fabricante Research In Motion.

Por último, se hace un análisis de identificación de modificaciones realizadas a las imágenes comparando la imagen en miniatura almacenada en la información Exif con el thumbnail generado a partir de la imagen. Este análisis se basa en el cálculo del RMS del resultado de la comparación

de las dos imágenes en miniatura. Los valores RMS procedentes de esta comparación se clasifican en 3 grupos: "no modificadas" para aquellas imágenes que tengan un valor RMS inferior a 5, "posiblemente modificadas" para aquellas imágenes con valor RMS entre 5 y 25, y "modificadas" para aquellas imágenes con valor RMS superior a 25. El resultado de este análisis se puede observar en la Tabla II.

Tabla II
RESULTADOS DEL ANÁLISIS DE IMÁGENES MODIFICADAS.

Clasificación	Observaciones	Subtotal	Total
Sin imagen en miniatura		872	872
No modificadas	Imagen rotada	5	322
	Diferente tamaño	131	
	Mismo tamaño y orientación	186	
Posiblemente modificadas	Imagen rotada	26	2335
	Diferente tamaño	428	
	Mismo tamaño y orientación	1881	
Modificadas	Imagen rotada	2	222
	Diferente tamaño	100	
	Mismo tamaño y orientación	120	
Total		3751	3751

V. CONCLUSIONES

En este trabajo se ha desarrollado una herramienta específica para el análisis forense de imágenes digitales de dispositivos móviles denominada *Theia* que contiene diferentes técnicas enfocadas a la identificación de la fuente de adquisición de imágenes digitales. Esta herramienta permite el tratamiento automático de los distintos conjuntos de metadatos Exif de imágenes digitales tanto individualmente como en grupo. Asimismo, integra dos técnicas de identificación de la fuente de adquisición de imágenes digitales: Una técnica basada en las características de la imagen que se apoya en la máquina de soporte vectorial para realizar la clasificación, y una técnica de clasificación sin supervisión de imágenes. Por último, permite la detección de manipulaciones basadas en el procesado de la imagen en miniatura. Seguidamente, se ha realizado un análisis de los metadatos Exif de un banco de imágenes de dispositivos móviles teniendo en cuenta no sólo los aspectos de obtención de la fuente de adquisición de la imagen. En general, con respecto a los metadatos Exif, puede concluirse que son de gran utilidad, ya que los fabricantes los insertan en el proceso de generación de la imagen. Asimismo, ha podido comprobarse que existen metadatos que pueden aportar información relevante al analista forense como son los datos de geoposicionamiento y la imagen en miniatura. Finalmente, *Theia* se ha comparado con diversas herramientas con fines similares concluyéndose que, desde el punto de vista del análisis forense, *Theia* es la única que ofrece funcionalidades avanzadas que permiten trabajar con proyectos con numerosas imágenes para identificación de la fuente de adquisición.

AGRADECIMIENTOS

Los autores agradecen la financiación que les brinda el Programa Marco de Investigación e Innovación Horizonte 2020 de la Comisión Europea a través del Proyecto H2020-FCT-2015/700326-RAMSES (Internet Forensic Platform for Tracking the Money Flow of Financially-Motivated Malware).

REFERENCIAS

[1] T. Ahonen and A. Moore, "Tomi Ahonen Almanac 2014: Mobile Telecoms Industry Annual Review," http://goo.gl/B1eX8, 2014.
[2] "Embedded Imaging Takes Off as Stand-alone Digital Cameras Stall," 2013. [Online]. Available: http://goo.gl/WnIiM5
[3] "Alexa Top 500 Global Sites," http://www.alexa.com/topsites, 2014.
[4] M. Al-Zarouni, "Mobile Handset Forensic Evidence: a Challenge for Law Enforcement," in *Proceedings of the 4th Australian Digital Forensics Conference*, Perth Western, Australia, December 2006, pp. 1–10.
[5] C. Y. Wen and K. T. Yang, "Image Authentication for Digital Image Evidence," *Forensic Science Journal*, vol. 5, no. 1, pp. 1–11, 2006.
[6] V. L. L. Thing, K. Y. Ng, and E. C. Chang, "Live Memory Forensics of Mobile Phones," *Digital Investigation*, vol. 7, pp. 74–82, August 2010.
[7] T. Gloe, M. Kirchner, A. Winkler, and R. Bohme, "Can We Trust Digital Image Forensics?" in *Proceedings of the 15th International Conference on Multimedia*, Augsburg, Germany, September 2007, pp. 78–86.
[8] C. W. Hsuand, C. C. Chang, and C. J. Lin, "A Practical Guide to Support Vector Classification," Department of Computer Science and Information Engineering, National Taiwan University," Practical Guide, April 2003.
[9] T. Van Lanh, K. S. Chong, S. Emmanuel, and M. S. Kankanhalli, "A Survey on Digital Camera Image Forensic Methods," in *Proceedings of the IEEE International Conference on Multimedia and Expo*, Beijing, China, July 2007, pp. 16–19.
[10] S. Committee, "Exchangeable Image File for digital still cameras: Exif version 2.3, April 26, 2010," http://goo.gl/jgrCpC, 2013.
[11] R. Baer, "Resolution Limits in Digital Photography: The Looming End of the Pixel Wars," in *Proceedings of the Imaging Systems Conference*, Tucson, Arizona United States, June 2010.
[12] A. L. Sandoval Orozco, D. M. Arenas González, L. J. García Villalba, and J. C. Hernández-Castro, "Anomalías en el Seguimiento de Exif en el Análisis Forense de Metadatos de Imágenes de Móviles," in *Actas del XII Reunión Española sobre Criptología y Seguridad de la Información*, Donostia-San Sebastián, España, Septiembre 2012.
[13] A. L. Sandoval Orozco, D. M. Arenas González, L. J. García Villalba, and J. C. Hernández Castro, "Analysis of Errors in Exif Metadata on Mobile Devices," *Multimedia Tools and Applications*, vol. 68, no. 1, pp. 1–29, January 2014.
[14] M. J. Tsai, C. L. Lai, and J. Liu, "Camera/Mobile Phone Source Identification for Digital Forensics," in *Proceedings of the International Conference on Acoustics Speech and Signal Processing*, Honolulu, Hawai, USA, April 2007, pp. 221–224.
[15] C. Mckay, A. Swaminathan, H. Gou, and M. Wu, "Image Acquisition Forensics: Forensic Analysis to Identify Imaging Source," in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing*, Las Vegas, Nevada, USA, June 2008, pp. 1657–1660.
[16] B. Wang, Y. Guo, X. Kong, and F. Meng, "Source Camera Identification Forensics Based on Wavelet Features," in *Proceedings of the International Conference on Intelligent Information Hiding and Multimedia Signal Processing*, Kyoto, Japan, September 2009, pp. 702–705.
[17] Y. Hu, C.-T. Li, and C. Zhou, "Selecting Forensic Features for Robust Source Camera Identification," in *Proceedings of the International Computer Symposium*, Tainan, China, December 2010, pp. 506–511.
[18] A. L. Sandoval Orozco, D. M. Arenas González, J. Rosales Corripio, L. J. García Villalba, and J. C. Hernandez-Castro, "Source identification for mobile devices, based on wavelet transforms combined with sensor imperfections," *Computing*, vol. 96, no. 9, pp. 829–841, 2014.
[19] J. Lukas, J. Fridrich, and M. Goljan, "Digital Camera Identification from Sensor Pattern Noise," *IEEE Transactions on Information Forensics and Security*, vol. 1, no. 2, pp. 205–214, June 2006.
[20] L. J. García Villalba, A. L. Sandoval Orozco, and J. Rosales Corripio, "Smartphone Image Clustering," *Expert Systems with Applications*, vol. 42, no. 4, pp. 1927–1940, 2015.

Uso de Características en la Identificación de la Fuente de Imágenes de Dispositivos Móviles

Jocelin Rosales Corripio, Esteban Alejandro Armas Vega, Ana Lucila Sandoval Orozco, Luis Javier García Villalba
Grupo de Análisis, Seguridad y Sistemas (GASS)
Departamento de Ingeniería del Software e Inteligencia Artificial
Facultad de Informática, Despacho 431, Universidad Complutense de Madrid (UCM)
Calle Profesor José García Santesmases, 9, Ciudad Universitaria, 28040 Madrid
Email: {jocelinr, esarmas}@ucm.es, {asandoval, javiergv}@fdi.ucm.es

***Resumen*—El análisis forense de imágenes digitales cobra especial importancia en la actualidad, dado el alto uso de las cámaras digitales de los dispositivos móviles. La identificación del tipo de dispositivo o la marca y modelo de la fuente de la imagen son dos ramas relevantes del análisis forense de imágenes digitales. En este trabajo se han tratado ambas, con un enfoque basado en distintos tipos de características de la imagen y clasificación mediante máquinas de soporte vectorial. Principalmente el estudio se ha centrado en imágenes generadas con dispositivos móviles, Por tanto, las técnicas y características han sido adaptadas o creadas con este fin. Se han realizado un total de 36 experimentos clasificados en 5 conjuntos, con el objetivo de probar distintas configuraciones posibles de las técnicas. Los resultados obtenidos son satisfactorios en todos los experimentos realizados superando en tasa de acierto a otras propuestas descritas en el estado del arte.**

***Palabras clave*—Análisis forense, color de la imagen, métricas de calidad de la imagen, PRNU, wavelet, identificación de la fuente de adquisición (Forensics Analysis, image quality metrics, PRNU, wavelet, image source adquisition identification)**

I. Introducción

Actualmente, la demanda de dispositivos móviles (teléfonos móviles, smartphones, tablets, etc.) aumenta año tras año a pesar de la situación de crisis económica global. Según Gartner [1] en 2013 las ventas de smartphones crecieron un 42.3 % respecto al año anterior, superando por primera vez en número de ventas a los teléfonos móviles tradicionales. Según [2], en 2016 de cuota de ventas de cámaras digitales en dispositivos móviles será del 48 %, siendo en este mismo año unicamente un 27 % la cuota de mercado de las DSC. Asimismo, existen predicciones que indican que las DSC desaparecerán en pro de las nuevas integradas en dispositivos móviles [3], ya que el aumento de calidad de estas crece a un ritmo imparable. No debemos dejar pasar por alto la irrupción en la sociedad actual de este tipo de dispositivos en nuestro día a día. El incremento de las capacidades de almacenamiento, procesamiento y usabilidad de los dispositivos móviles, así como el coste asequible de los mismos, han permitido que estén presentes en diversas actividades, lugares y eventos de la vida cotidiana. Tanto es así que, según [4], un gran número de personas tienen y usan más de un dispositivo móvil. Asimismo, un usuario típico en promedio consulta su móvil unas 150 veces al día y más de 90 % de las personas que alguna vez han tomado una fotografía lo ha hecho únicamente con cámaras de dispositivos móviles.

A causa del uso extenso de cámaras digitales en dispositivos móviles se han generado polémicas, discusiones y normas sobre la prohibición de su uso en lugares como oficinas gubernamentales, colegios, empresas, etc. Una consecuencia de su extenso uso es que las imágenes digitales pueden ser utilizadas como testigos silenciosos en procesos judiciales (pornografía infantil, espionaje industrial, ...), siendo en muchos casos piezas cruciales de la evidencia de un crimen [5]. Por estas razones el análisis forense de imágenes digitales de dispositivos móviles cobra especial fuerza en la actualidad.

El análisis forense de imágenes digitales se puede dividir en dos grandes ramas [6]: autenticidad de las imágenes e identificación de la fuente de adquisición de la imagen. La primera de las ramas tratar de discernir si una imagen ha sufrido algún procesamiento posterior al de su creación, es decir, que no haya sido manipulada. La segunda de las ramas, que será tratada en este trabajo, tiene como fin identificar el tipo (cámara, escáner, computador,...) o clase (marca y modelo) de la fuente de adquisición de la imagen digital.

Este trabajo está estructurado en 5 secciones, siendo el primero la presente introducción. En la sección II se expone el estado del arte de técnicas y algoritmos para la identificación del tipo de fuente y la identificación de la fuente de adquisición. La sección III describe los distintos conjuntos de características (Noise, Color, Image Quality Metrics (IQM) y Wavelets) utilizadas en la técnica de identificación de la fuente de adquisición propuesta. En la sección IV se realizan un conjunto de experimentos para la identificación del tipo de dispositivo y la identificación de la fuente de adquisición de la imagen. Por último, en la sección V presentan las principales conclusiones extraídas de este trabajo.

II. Técnicas de Análisis Forense en Imágenes

En esta sección se describen las principales técnicas de análisis forense de imágenes digitales para la identificación de la fuente de adquisición de la imagen y los principales trabajos del estado del arte. El éxito de estas técnicas depende del supuesto de que todas las imágenes adquiridas por un mismo dispositivo presentan características intrínsecas del mismo. Las

características que se usan para identificar marca y modelo de las cámaras digitales se derivan de las diferencias que existen entre las técnicas de procesamiento de las imágenes y las tecnologías de los componentes que se utilizan. El mayor problema con este enfoque es que los diferentes modelos de las cámaras digitales usan componentes de un número reducido de fabricantes, y que los algoritmos que usan también son muy similares entre modelos de la misma marca. Según [7] se pueden establecer cuatro grupos de técnicas para este fin: basadas en la aberración de las lentes, basadas en la interpolación de la matriz CFA, basadas en el uso de las características de la imagen y las basadas en las imperfecciones del sensor. Además de los anteriores grupos existe otro grupo de técnicas forenses a destacar basadas en los metadatos de la imagen.

Las técnicas basadas en los metadatos de la imagen son las más sencillas. Existen gran cantidad de trabajos enfocados sobre los diferentes tipos de metadatos, tanto para la búsqueda de información, como para la clasificación de imágenes [8–10]. Asimismo los metadatos pueden utilizarse como datos de entrada o ayuda para el uso de otras técnicas forenses. Sin embargo, estas técnicas dependen en gran medida de los metadatos que los fabricantes deciden insertar cuando la imagen es generada y en la corrección de los mismos. En [11, 12] se realiza un estudio a fondo sobre este tema. Asimismo, este método es el más vulnerable a modificaciones malintencionadas.

Las técnicas basadas en las características de las imágenes utilizan un conjunto de características extraídas del contenido de la imagen para hacer la identificación de la fuente. Estas características se dividen en cuatro grupos: características de color, métricas de calidad de la imagen (*Image Quality Metrics* IQM), estadísticas del dominio *wavelet* y características del ruido resultante de los defectos introducidos durante el proceso de generación de una imagen.

En [13] se propone un método de identificación de la fuente utilizando las siguientes características: color, calidad de la imagen y dominio de la frecuencia. En el estudio adoptan la transformada *wavelet* como método para calcular las estadísticas del dominio *wavelet* y utilizan SVM para la clasificación. En los experimentos realizados se usaron cámaras digitales y dispositivos móviles. Los resultados obtenidos en los distintos experimentos arrojan unos resultados entre el 61.7 % y el 99.72 % de acierto.

En [14] se extiende la identificación de la fuente a diferentes dispositivos tales como teléfonos móviles con cámara integrada, cámaras digitales, escáneres y computadoras. En esta propuesta se usan las diferencias en el proceso de adquisición de la imagen de los dispositivos para formar dos grupos de características: coeficientes de interpolación de color y características de ruido. En los experimentos se utilizaron cinco modelos de teléfonos móviles, cinco modelos de cámaras digitales y cuatro modelos de escáneres para identificar el tipo de fuente. En los resultados globales se obtuvo un 93,75 % de precisión. En el análisis de identificación de marca y modelo de teléfonos móviles obtuvieron una precisión del 97,7 % para los cinco modelos.

En [15] se propone un método para la identificación de la cámara fuente mediante la extracción y clasificación de las estadísticas de las características *wavelets*. Finalmente se obtiene 216 características *wavelet* de primer orden y 135 características de co-ocurrencia de segundo orden. Se seleccionan las características más representativas utilizando un algoritmo SFFS y se clasifican utilizando una SVM. Se consigue una media de identificación del 98 % para el conjunto de todas las cámaras y una tasa de acierto media del 96.9 % para las tres cámaras del mismo modelo.

En [16] se realizan experimentos con las características de las imágenes para la identificación de la fuente más usuales: *wavelet*, color, IQM, características estadísticas de diferencia de imágenes y características estadísticas de predicción de errores. En los experimentos se proponen distintas combinaciones de los distintos tipos de características y una SVM para la clasificación de los distintos dispositivos. Se utilizan diez cámaras diferentes de cuatro fabricantes distintos con 300 imágenes de cada cámara (150 para entrenamiento y 150 para testear) y de una resolución de 1024x1024. Utilizando todas las características se obtiene un resultado de acierto medio del 92 %.

En [17] se propone un método que emplea la densidad marginal de los coeficientes de la Transformada Coseno Discreta (DCT) en las coordenadas de frecuencia baja y las características de densidad del vecindario (*neighbouring joint density*) en el domino DCT. Adicionalmente, se utiliza la agrupación jerárquica (*hierarchical clustering*) y una SVM para detectar la fuente de adquisición de las imágenes. En los experimentos realizados con imágenes pertenecientes a cinco modelos de teléfonos inteligentes de cuatro fabricantes, se obtuvo entre el 86,36 % y el 99,91 % de acierto, alcanzando mejores resultados con un *kernel* SVM lineal.

En [18] se propone la combinación de dos técnicas (Imperfecciones del sensor y la transformada wavelet) para identificar la fuente de adquisición de la imagen generada con un dispositivo móvil. Este método extrae el patrón del ruido del sensor. Posteriormente, se obtienen 25 características (16 características de primer orden y alto orden y 9 característica aplicando filtros QMFs (*Separable Quadratic Mirror Filters*)).

Finalmente, las técnicas que estudian las huellas que los defectos del sensor pueden dejar sobre las imágenes se dividen en dos ramas: defectos de píxel y patrón de ruido del sensor (SPN). En la primera se estudian los defectos de píxel, píxeles calientes, píxeles muertos, defectos de fila o columna y los defectos de grupo. En la segunda se construye un patrón del ruido promediando los múltiples residuos de ruido obtenidos mediante algún filtro de eliminación de ruido. La presencia del patrón se determina utilizando algún método de clasificación como correlación o máquinas SVM.

En [19] se analiza el patrón de ruido del sensor de un conjunto de cámaras, el cual funciona como una huella dactilar, permitiendo la identificación única de cada cámara. Para obtener este patrón se realiza un promedio del ruido obtenido a partir de diferentes imágenes utilizando un filtro de eliminación de ruido. Para identificar la cámara a partir

de una imagen dada, se considera el patrón de referencia como una marca de agua cuya presencia en la imagen es establecida mediante un detector de correlación. Este método está afectado por algoritmos de procesamiento de la imagen, como la compresión JPEG y la corrección *gamma*. Según [7] los resultados para fotografías recortadas no son satisfactorios.

En [20] se propone un enfoque para la identificación fuente de la cámara considerando escenarios abiertos, donde a diferencia de los escenarios cerrados no se da por sentado contar con acceso a todas las posibles cámaras de origen de la imagen. Esta propuesta comprende tres fases: definición de las regiones de interés, determinación de las características e identificación de la cámara fuente. Para determinar las características se calcula el SPN para cada uno de los canales R, G, B e Y (luminancia), generándose un total de 36 características para representar cada imagen. En los experimentos se utiliza un 25 cámaras digitales de 9 fabricantes. Los resultados mostraron una precisión del 94,49 %, del 96,77 % y del 98,10 %.

III. Técnica de Identificación de la Fuente de Adquisición de una Imagen

La técnica para la identificación de la fuente de adquisición de una imagen (tipo de fuente o marca y modelo de la fuente) propuesta, está basada en la extracción de características del contenido de la propia imagen. El conjunto de características a utilizar puede clasificarse en cuatro grandes grupos, dependiendo de la naturaleza de la obtención de las mismas: características de ruido (16), características de color (12), métricas de calidad de la imagen (40) y *wavelets* (81). Para la identificación se utilizan 2 conjuntos de imágenes: imágenes de fuentes conocidas para entrenar el clasificador SVM [21] y otro conjunto de imágenes de fuentes desconocidas que se utilizarán en la fase de predicción para averiguar su fuente de adquisición. El *kernel* utilizado para clasificar es *Non-linear RBF*, dado que es recomendado cuando no se cuenta con la información apriori de los datos. A continuación se va a realizar un análisis detallado sobre cada uno de los conjuntos de características citados anteriormente.

III-A. Características del Ruido

El proceso de generación de imágenes suele introducir varios defectos en estas, los cuáles crearán ruido que aparecerá en la imagen final. Un tipo de este ruido, es causado por defectos de la matriz CFA (*hot point defects*, *dead pixels*, *pixel traps*, *column defects* y *cluster defects* entre otros). Este tipo de defectos causan que dichos pixeles difieran en gran medida de los restantes de la imagen original, siendo en muchos casos indiferente que se tenga una u otra imagen, ya que este pixel mostrará siempre el mismo valor. Existen distintos filtros para conseguir suavizar el efecto de este ruido. Por sencillez y velocidad se utilizará el filtro Gaussiano. Este filtro será usado para eliminar el ruido en las imágenes y posteriormente obtener las distintas características.

Uno de nuestros objetivos es conseguir un conjunto de características que nos permitan diferenciar entre los distintos tipos de dispositivos. Para ello, como primer paso, se tiene en cuenta que las cámaras digitales utilizan un sensor array bidimensional mientras que la mayoría de los escáneres utilizan un sensor array lineal. En el caso de los escáneres la misma disposición lineal del sensor se traslada para generar toda la imagen; Por tanto, se espera encontrar periodicidad del ruido del sensor entre las filas de la imagen escaneada. Asimismo, no hay razón para encontrar una periodicidad del ruido del sensor entre las columnas de la imagen escaneada. Para el caso de cámaras digitales este tipo de periodicidad del ruido no existe. Esta diferencia se puede utilizar como base para discriminar entre los distintos tipos de dispositivos.

El ruido de la imagen original puede ser modelado como la suma de dos componentes, el ruido constante y el ruido aleatorio . Para los escáneres el ruido constante solo depende del índice de la columna, ya que el mismo sensor es trasladado verticalmente para generar la imagen completa. La media del ruido de todas las columnas puede ser usada como patrón de referencia ya que las componentes aleatorias del ruido se anularan. Para detectar la similitud entre las diferentes filas con el patrón de referencia, se utilizará la correlación de estas con dicho patrón. Posteriormente, se realizará el mismo proceso para detectar la similitud de las columnas con el patrón de referencia. Finalmente, se obtienen el conjunto de características en sí. Cabe destacar a la hora de obtener las características, que la orientación de la imagen en el caso de los escáneres es fundamental, ya dependiendo de ésta las características obtenidas serán completamente diferentes.

Para cada tipo de correlación se obtienen valores estadísticos de primer orden: media, mediana, máximo y mínimo. Otras características de orden alto obtenidas son la varianza, kurtosis y skewness. Todas ellas miden valores estadísticos más específicos que las anteriores. Asimismo se añaden las características del ratio entre las correlaciones de filas y de columnas. Por último se incluyó la característica del ruido medio por pixel. Esta característica no depende de las correlaciones de filas o columnas con el patrón de referencia, sino que es independiente y permite distinguir entre los distintos tipos de dispositivos, como pueden ser las imágenes generadas por computador. En total se obtienen 16 características: 7 características de filas, 7 de columnas, el ratio entre las correlaciones de filas y de columnas y el ruido medio del pixel.

III-B. Características de Color

La configuración de los filtros de la matriz CFA, el algoritmo de demosaicing y la técnicas aplicadas al procesamiento del color hacen que las señales contenidas en la bandas de color tengan tratamientos y patrones específicos. Con el objetivo de determinar las diferencias en las características del color para los diferentes modelos de cámaras es necesario examinar las estadísticas de primer y segundo orden de las imágenes tomadas con ellas. A continuación se proponen un conjunto de 12 características de color:

- Valor medio de los píxeles: El promedio de los valores de los canales RGB de una imagen debe dar como

resultado el color gris, siempre y cuando la imagen tenga suficientes variaciones de color (3 características).
- Correlación de los pares RGB: Dependiendo de la estructura de la cámara la correlación entre las diferentes bandas de color puede variar. Utilizamos el coeficiente correlación de Pearson para determinar la correlación entre las bandas RG, RB y GB (3 características).
- Distribución de vecindad del centro de masa: Primero se calcula el número total de píxeles para cada valor de color. Después, se obtienen la suma de los valores vecinos. Por último, se calcula el centro de masa de este último vector, lo que va a devolver un valor entre 0 y 255 (3 características).
- Ratio de la energía entre los pares RGB: Esta característica depende del proceso de corrección de puntos blancos que realiza la cámara (3 características).

III-C. Métricas de Calidad de la Imagen

Los diferentes modelos de cámara producen imágenes de diferente calidad. Puede haber diferencias en la luminosidad de la imagen, la nitidez o en la calidad del color. Estas diferencias hacen que se proponga un conjunto de métricas de calidad, como características que nos ayudan a diferenciar la fuente de las imágenes. Existen diferentes categorías para estas métricas: las medidas basadas en las diferencia de los píxeles, las medidas basadas en la correlación y las medidas basadas en la distancia espectral.

Para obtener este conjunto de métricas se necesita la imagen original y una imagen filtrada en la que se reduzca el ruido de la imagen original. Se utiliza un filtrado gaussiano que nos permite llevar a cabo el suavizado de la imagen. Para obtener las métricas se utiliza un filtro con un núcleo de tamaño 3x3 con $\sigma = 0{,}5$. Cada píxel de la nueva imagen se obtiene realizando la transformación de vecindad sobre el píxel de la imagen original utilizando el núcleo anteriormente calculado. Es necesario tener en cuenta los bordes de la imagen al realizar la transformación. En nuestro caso se ha optado por considerar un borde exterior con píxeles de valor 0. En total se extraen 40 características, entre las que se destacan las siguientes:

- Distancia Czekonowsky: Útil para comparar vectores con componentes no negativas como es el caso de las imágenes en color.
- Métricas de Minkowsky: Para $\gamma = 1$ y $\gamma = 2$ se calcula la norma L_γ de disimilitud entre dos imágenes, donde N^2 es el número total de píxeles. $\gamma = 1$ se corresponde con el *Mean Absolute Error* (MAE) y $\gamma = 2$ con el *Mean Square Error* (MSE). En ambos casos, valores elevados de MAE o MSE se corresponden con imágenes de baja calidad.
- *Laplacian Mean Square Error* (LMSE): Está basada en la importancia de la medición de los bordes. Un valor alto del LMSE indica que la imagen es de baja calidad.
- Cross-Correlación Normalizada: La cercanía entre dos imágenes digitales también puede ser cuantificada en términos de una función de correlación.
- Métrica de calidad del contenido estructural de una imagen definida para cada banda.
- Medidas Espectrales: Se calcula la transformada discreta de Fourier (DFT) de las imágenes original y suavizada para definir las siguientes métricas de calidad para cada banda: Fase espectral, Magnitud espectral y una media ponderada entre la fase y la magnitud espectral. Estas características también se pueden obtener para cada bloque de la imagen. Portanto, se divide la imagen en L bloques de tamaño bxb, y luego se calculan dichas características (se ha utilizado $\gamma = 2$ y un tamaño de bloque de 32x32). Posteriormente, para cada bloque se obtienen las siguientes características: *Median Block Spectral Magnitude*, *Median Block Spectral Phase*, *Median Block Weighted Spectral Distance*.
- Medidas basadas en el sistema visual humano (HVS): Las imágenes pueden ser procesadas mediante filtros que simulan el HVS. Uno de los modelos utilizados para ello es un filtro pasa banda con una función de transferencia en coordenadas polares. las métricas de calidad que obtenemos para cada banda de la imagen basadas en estas medidas son: *Normalized absolute error* y *HVS based L2*.

III-D. Características Wavelet

Debido a la propiedad determinista del patrón de ruido del sensor presente en una imagen, este patrón puede usarse como huella para identificar el dispositivo que generó la imagen. Así, para clasificar e identificar una imagen se requiere de un algoritmo que nos permita extraer el ruido del sensor y otro que nos permita obtener las características de las huellas obtenidas. Para la extracción del ruido del sensor se utiliza el algoritmo presentado en [22].

Finalmente, con el algoritmo 1 se calculan un total de 81 características (3 canales x 3 componentes *wavelet* x 9 momentos centrales).

Algorithm 1: Extracción de Características

Input: Imagen
Huella del sensor de la imagen
Result: 81 características

1 **procedure** EXTRAERCARACTERISTICAS(I)
2 Separar los canales R, G y B de la huella del sensor;
3 **foreach** *canal de color* **do**
4 Hacer una descomposición wavelet de un nivel;
5 **foreach** $c \in \{H,V,D\}$ **do**
6 Calcular k momentos centrales con $m_k = \frac{1}{n}\sum_{i=1}^{n} |c_i - \overline{c}|^k$;
7 **end procedure**

IV. EXPERIMENTOS Y RESULTADOS

Para la evaluación de la propuesta se realizaron dos tipos de experimentos: identificación del tipo de dispositivo fuente e

identificación de la fuente de adquisición. En los experimentos se realizan sobre 200 imágenes de cada tipo de dispositivo, 100 para la fase de entrenamiento y 100 para la fase de predicción. Todas las imágenes tienen una resolución mayor a 1024x768.

IV-A. Identificación del Tipo de Dispositivo Fuente

En el experimento se uso el siguiente dataset: imágenes de 12 teléfonos móviles, imágenes obtenidas de 15 escáneres e imágenes generadas digitalmente por un computador. Las imágenes de escáneres y las generadas por ordenador se descargaron de *Flickr*. En los experimentos se han tenido en cuenta los siguientes parámetros de configuración: tamaño de recorte aplicado, posición del recorte (centrado (C) o a la esquina superior izquierda (ESI)) y aplicación de distintos conjuntos de características (ruido, color, IQM y wavelet). La Tabla I muestra los resultados obtenidos en 10 experimentos.

Tabla I
TASAS DE ACIERTO DEL TIPO DE DISPOSITIVO FUENTE

C\ticas	Recorte	Tipo de recorte	Dispositivo			% de acierto
			Cámara	Ordenador	Escáner	
Ruido	No	-	70	54	57	**59,95**
Ruido	1024x768	C	66	80	46	**62,39**
Ruido	800x600	C	76	60	49	**60,68**
Ruido	640x480	C	62	61	48	**56,62**
Ruido	1024x768	ESI	76	59	40	**56,40**
Ruido	800x600	ESI	65	38	44	**47,72**
Ruido	640x480	ESI	74	54	37	**52,88**
Todas	1024x768	C	66	73	72	**70,26**
Todas	800x600	C	69	74	71	**71,30**
Todas	640x480	C	77	73	63	**70,75**
	Promedio		**69,91**	**61,35**	**51,42**	**60,42**

En la tabla se observa que las tasas de acierto no son muy altas (un 60,42 % de media y un 71,30 % en el mejor de los casos). El alto número de clases de dispositivos (marcas y modelos) de los diferentes tipos dificulta la clasificación resultando tasas de acierto bajas. Sin embargo, sí arrojan información sobre los parámetros de configuración utilizados, ya que entre el mejor y el peor resultado hay una diferencia de 23,48 % de tasa de acierto. En general, el uso de una única de las características del ruido no da buenos resultados para la identificación del tipo de fuente cuando el número de dispositivos a clasificar es alto, ya que la tasa media de acierto de todos los experimentos es del 56,65 %. Asimismo, Los resultados mejoran notablemente cuando se utilizan todas las características para la identificación del tipo de fuente. Dado el alto número de clases los resultados son aceptables con un 70,77 % de tasa de acierto en promedio. Adicionalmente se observa que el tamaño de recorte afecta a los resultados: a menor tamaño de recorte menor tasa de acierto. Con un tamaño de recorte 1024x768 se obtienen mejores resultados que cuando se utiliza todo el tamaño de la imagen.La Tabla II muestra la tasa de acierto cuando el dataset se reduce a 2 tipos de dispositivos, se utilizan únicamente las características de ruido, y la posición de recorte usada es centrada.

Se observa que cuando el número de tipos de dispositivos se reduce a dos los resultados son aceptables (la tasa de aciertos sube a 85,44 % de media). Se observa que el tamaño del recorte no afecta significativamente a los resultados. Los mejores resultados obtenidos son los que distinguen entre imágenes de teléfonos móviles y escáneres (96,23 % de media). El segundo mejor resultado se presentra entre imágenes de escáneres y las generadas por ordenador (81,62 % de media). El peor de los resultados se presenta entre imágenes generadas por ordenador y Teléfonos móviles (79,42 % de media).

Tabla II
USO DE C\TICAS DE RUIDO EN LA IDENTIFICACIÓN

Tipo de Dispositivo		Tamaño de recorte	% de acierto
1	2		
Escáner	Teléfono móvil	1024x768	95,79
Escáner	Teléfono móvil	640x480	96,16
Escáner	Teléfono móvil	400x300	96,73
Ordenador	Teléfono móvil	1024x768	79,96
Ordenador	Teléfono móvil	640x480	79,76
Ordenador	Teléfono móvil	400x300	78,55
Ordenador	Escáner	1024x768	82,87
Ordenador	Escáner	640x480	81,10
Ordenador	Escáner	400x300	80,91

IV-B. Identificación de la Fuente de la Imagen en Dispositivos Móviles

Dada la importancia de las imágenes de los dispositivos móviles en la actualidad, se realizaron experimentos para identificar la marca y modelo de la cámara que las generó. Se utilizaron 7 modelos de teéfonos móvviles: IPhone 4S, BB 8520, Huawei U8815, LG E400, Nokia 800, Samsung GT-I9001 y Sony Ericsson C2105. En el experimento se analiza el uso de los diferentes tipos de características y el tamaño de recorte. La Tabla III muestra los resultados utilizando diferentes tipos de combinaciones de las características.

Tabla III
IDENTIFICACIÓN DE LA FUENTE PARA 7 DISPOSITIVOS MÓVILES

C\ticas	Recorte	I1	HU	LG2	N1	BB	S1	SE1	% de acierto
Todas	1024x768	93	96	80	94	91	70	85	86,54
Ruido	1024x768	41	42	35	18	40	40	62	37,67
Color	1024x768	24	37	20	40	31	19	44	29,27
IQM	1024x768	13	88	46	89	7	34	2	21,65
Wavelet (a)	1024x768	95	96	96	94	92	76	93	91,46
Wavelet (b)	1024x768	95	87	97	70	86	56	91	81,84
Color+IQM +Wavelet	1024x768	93	94	90	90	90	53	85	83,67
Todas	800x600	91	96	84	92	95	56	85	84,41
Todas	640x480	90	95	84	89	88	51	88	82,15

El experimento revela que las características de ruido, color e IQM por separado son inválidas, ya que a lo sumo se obtiene una tasa media de acierto del 37,67 % siendo inaceptable. Con las wavelets se realizaron dos experimentos utilizando distintos

tipos de wavelet: (a) Daubechies 8-tap y (b) Haar. Los resultados muestran que Daubechies 8-tap obtiene mejores resultados que Haar y a su vez los mejores resultados (91,46 %). Con respecto a las distintas combinaciones de características, se observa que al utilizar todas las características se obtienen buenos resultados (86,54 %), ya que aunque son ligeramente peores que el mejor resultado, la diferencia no es muy alta (un 4,92 %). Asimismo, puede verse que la tasa de acierto usando todas las características baja sutilmente cuanto más pequeño es el tamaño del recorte. La combinación de todas las características menos las de ruido obtiene una tasas media de acierto del 83,67 %. Estos resultados, no siendo malos, distan del obtenido con las wavelets y son peores que cuando se utiliza la combinación de todas las características.

V. Conclusiones

En este trabajo se presenta una técnica para la identificación de imágenes de dispositivos móviles entre imágenes de escaneres y generadas por ordenador. Asimismo, permiten distinguir la fuente de adquisición de imágenes de dispositivos móviles. La técnica se basa en el uso de cuatro conjuntos de características (ruido, color, IQM y wavelets). Se ha experimentado con la combinación de los distintas de características, tamaños y posición de recorte y funciones wavelets. En la identificación del tipo de fuente se descartan las características de ruidos como inválidas cuando el número de tipos de dispositivos es mayor que 2. Esto se debe a que en los experimentos se obtuvieron resultados inaceptables en la identificación con tres tipos de dispositivos (escáner, dispositivo móvil y ordenador). Como contrapartida, los analistas forenses si pueden tener en cuenta la aplicación de la técnica con las características de ruido para la identificación del tipo de fuente de imágenes de dispositivos móviles con respecto a imágenes de escáneres y ordenadores. En la identificación de la fuente de adquisición de la imagen de dispositivos móviles, los resultados son mucho mejores. En los experimentos realizados, al menos hay una configuración que arroja buenos resultados, siempre contextualizándolos en el nivel de exigencia que se hace sobre la técnica. El uso de todas las características o las wavelets con Daubechies 8-tap son las que ofrecen mejores resultados. Con respecto al tamaño de recorte, para la obtención de resultados óptimos existe un tamaño de recorte óptimo (1024x768), que no necesariamente es el mayor o la imagen entera, ya que esta última genera peores resultados que cuando se utiliza un recorte.

Agradecimientos

Los autores agradecen la financiación que les brinda el Programa Marco de Investigación e Innovación Horizonte 2020 de la Comisión Europea a través del Proyecto H2020-FCT-2015/700326-RAMSES (Internet Forensic Platform for Tracking the Money Flow of Financially-Motivated Malware).

Referencias

[1] G. Inc., "Gartner Says Annual Smartphone Sales Surpassed Sales of Feature Phones for the First Time in 2013," http://goo.gl/5zp8ki, 2014.

[2] "Embedded Imaging Takes Off as Stand-alone Digital Cameras Stall," 2013. [Online]. Available: http://goo.gl/WnIiM5

[3] R. Baer, "Resolution Limits in Digital Photography: The Looming End of the Pixel Wars," in *Proceedings of the Imaging Systems Conference*, Tucson, Arizona United States, June 2010.

[4] T. Ahonen and A. Moore, "Mobile Telecoms Industry Annual Review," http://goo.gl/v9enrF, 2012.

[5] M. Al-Zarouni, "Mobile Handset Forensic Evidence: a Challenge for Law Enforcement," in *Proceedings of the 4th Australian Digital Forensics Conference*, Perth Western, Australia, December 2006, pp. 1–10.

[6] T. Gloe, M. Kirchner, A. Winkler, and R. Bohme, "Can We Trust Digital Image Forensics?" in *Proceedings of the 15th International Conference on Multimedia*, Augsburg, Germany, September 2007, pp. 78–86.

[7] T. Van Lanh, K. S. Chong, S. Emmanuel, and M. S. Kankanhalli, "A Survey on Digital Camera Image Forensic Methods," in *Proceedings of the IEEE International Conference on Multimedia and Expo*, Beijing, China, July 2007, pp. 16–19.

[8] M. Boutell and J. Luo, "Beyond Pixels: Exploiting Camera Metadata for Photo Classification," *Pattern Recognition*, vol. 38, no. 6, pp. 935–946, June 2005.

[9] N. L. Romero, V. G. Chornet, J. S. Cobos, A. S. Carot, F. C. Centellas, and M. C. Mendez, "Recovery of Descriptive Information in Images From Digital Libraries by Means of EXIF Metadata," *Library Hi Tech*, vol. 26, no. 2, pp. 302–315, 2008.

[10] D. M. Arenas González, "Análisis Forense de Imágenes de Móviles Mediante el Uso de Metadatos," Universidad Complutense de Madrid, Tesis de Máster 13507, Noviembre 2011.

[11] A. L. Sandoval Orozco, D. M. Arenas González, L. J. García Villalba, and J. C. Hernández-Castro, "Anomalías en el Seguimiento de Exif en el Análisis Forense de Metadatos de Imágenes de Móviles," in *Actas del XII Reunión Española sobre Criptología y Seguridad de la Información*, Donostia-San Sebastián, España, Septiembre 2012.

[12] A. L. Sandoval Orozco, D. M. Arenas González, L. J. García Villalba, and J. Hernández-Castro, "Analysis of errors in exif metadata on mobile devices," *Multimedia Tools and Applications*, pp. 1–29, 2014.

[13] M. J. Tsai, C. L. Lai, and J. Liu, "Camera/Mobile Phone Source Identification for Digital Forensics," in *Proceedings of the International Conference on Acoustics Speech and Signal Processing*, Honolulu, Hawai, USA, April 2007, pp. 221–224.

[14] C. Mckay, A. Swaminathan, H. Gou, and M. Wu, "Image Acquisition Forensics: Forensic Analysis to Identify Imaging Source," in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing*, Las Vegas, Nevada, USA, June 2008, pp. 1657–1660.

[15] B. Wang, Y. Guo, X. Kong, and F. Meng, "Source Camera Identification Forensics Based on Wavelet Features," in *Proceedings of the International Conference on Intelligent Information Hiding and Multimedia Signal Processing*, Kyoto, Japan, September 2009, pp. 702–705.

[16] Y. Hu, C.-T. Li, and C. Zhou, "Selecting Forensic Features for Robust Source Camera Identification," in *Proceedings of the International Computer Symposium*, Tainan, China, December 2010, pp. 506–511.

[17] Q. Liu, X. Li, L. Chen, H. Cho, A. P. Cooper, Z. Chen, M. Qiao, and A. H. Sung, "Identification of Smartphone-Image Source and Manipulation," in *Proceedings of the 25th International Conference on Industrial Engineering and Other Applications of Applied Intelligent Systems*, Dalian, China, June 2012.

[18] A. L. Sandoval Orozco, D. M. Arenas González, J. Rosales Corripio, L. J. García Villalba, and J. C. Hernandez-Castro, "Source identification for mobile devices, based on wavelet transforms combined with sensor imperfections," *Computing*, vol. 96, no. 9, pp. 829–841, 2014.

[19] J. Lukas, J. Fridrich, and M. Goljan, "Digital Camera Identification from Sensor Pattern Noise," *IEEE Transactions on Information Forensics and Security*, vol. 1, no. 2, pp. 205–214, June 2006.

[20] F. D. O. Costa, M. Eckmann, W. J. Scheirer, and A. Rocha, "Open Set Source Camera Attribution," in *Proceedings of the 25th Conference on Graphics, Patterns and Images*, Brazil, August 2012, pp. 71–78.

[21] C. W. Hsuand, C. C. Chang, and C. J. Lin, "A Practical Guide to Support Vector Classification," Department of Computer Science and Information Engineering, National Taiwan University," Practical Guide, April 2003.

[22] J. Rosales Corripio, D. M. Arenas González, A. L. Sandoval Orozco, L. J. García Villalba, J. C. Hernandez-Castro, and S. J. Gibson, "Source Smartphone Identification Using Sensor Pattern Noise and Wavelet Transform," in *Proceedings of the 5th International Conference on Imaging for Crime Detection and Prevention*, London, UK, December 2013, pp. 1–6.

Computing
DOI 10.1007/s00607-015-0482-5

Theia: a tool for the forensic analysis of mobile devices pictures

Ana Lucila Sandoval Orozco[1] · Jocelin Rosales Corripio[1] · David Manuel Arenas González[1] · Luis Javier García Villalba[1] · Julio Hernandez-Castro[2]

Received: 4 August 2014 / Accepted: 22 December 2015

Abstract Currently the number of cameras embedded in mobile devices is growing at an unprecedented rate. Additionally, the quality and performance of these mobile cameras is steadily improving, and is closing in on that of classical digital cameras. This scenario makes the forensic analysis of images taken with mobile cameras increasingly important and necessary. Among the various branches of forensic analysis, this paper focuses on the reliable acquisition of the make and model of the mobile camera that produced a given image. For this we have developed a technique based on exchangeable image file format (Exif) metadata analysis, allowing us in certain cases to obtain both the make and model with which the photo was taken. This comes with considerable analysis of whether this metadata information could be trusted, and with additional tools that can help in discovering image manipulation. These and other capabilities have been integrated into a new tool we have developed called *Theia*, that also offers many other advantages to the forensic analyst that has to mass process and analyze thousands of images in the fastest and most forensically sound way. To that end, we have also incorporated various complex functions that greatly help the forensic analyst, such as different types of advanced queries on Exif metadata information of large sets of images, and advanced geopositioning capabilities.

✉ Luis Javier García Villalba
javiergv@fdi.ucm.es

1 Group of Analysis, Security and Systems (GASS), Department of Software Engineering and Artificial Intelligence (DISIA), Faculty of Information Technology and Computer Science, Universidad Complutense de Madrid (UCM), Office 431, Calle Profesor José García Santesmases s/n, Ciudad Universitaria, 28040 Madrid, Spain

2 School of Computing, University of Kent, Office S129A, Cornwallis South Building, Canterbury CT2 7NF, UK

Published online: 25 January 2016

Keywords Mobile phone camera · Exif · Forensics analysis · Source identification

Mathematics Subject Classification 68Q32 Computational learning theory [See also 68T05] · 68T05 Learning and adaptive systems · 68U10 Image processing

1 Introduction

Nowadays, despite suffering the impact of global financial crisis, the sales of mobile devices such as smartphones or tablets, is still increasing. About 78.1 % of mobile phones sold in 2010 have an integrated camera [1]. Integrated cameras in mobile devices outnumber traditional Digital Still Cameras(DSCs). The sales of cameras integrated into mobile devices in 2013 exceeded 1.8 billion units. Similarly, there are predictions that the DSCs will disappear in favour of integrated mobile devices [2], since the quality of these cameras is growing at an unstoppable rate.

Also, the emergence of cameras in mobile devices should not only be measured in sales figures, as in our daily life it is common to see how people use photographs from these devices for a variety of situations—personal life, news, legal evidence, software applications and so on.

Many believe these cameras facilitate the proliferation of crimes related to privacy and information security (credit card theft, child pornography, industrial espionage, etc.). In fact, one of the main reasons for the existence of devices without cameras is that many companies, organizations or governments have rules that prohibit or limit its use [3]. On the other hand, there are reasons to justify that this vast expansion of cameras in mobile devices is beneficial to different situations which require graphic proof of a certain fact, such as criminal evidence, deprivation on liberty of the press as well as others. Therefore, it is necessary to provide forensic analysts with tools to facilitate its work for all kinds of investigations. Given the particular technical characteristics of this kind of photographs, forensic analysis tools should be specific, whilst those which deal with images not generated by mobile devices proving to be invalid.

The rest of the paper is structured into seven sections. Section 2 carries out a state of the art investigation into forensic analysis of images generated by mobile devices initially explaining the process of image acquisition in digital cameras and in mobile devices, and summarises the main techniques used with their respective results. Section 3 provides a description of the main metadata systems in images emphasizing the Exif standard for its high level of use on mobile devices generated images. Section 4 makes a binary analysis of real image metadata of several mobile phones. This analysis allows for a deeper understanding of the Exif standard, analyzing the Exif specification compliance by manufacturers, some of the violations found, and its consequences. Section 5 introduces the tool called *Theia* to be presented in this work. Firstly, a description of the functionalities is presented to later compare *Theia* to similar ones. Section 6 contains an important analysis over a collection of our own images dataset taken from different mobile devices. Lastly, Sect. 7 reports the results and the conclusions gathered.

2 Previous works in image forensic analysis

2.1 Image formation in digital cameras

The first step is to understand and create image processing forensic algorithms and to thoroughly know the process of image acquisition in digital cameras, which can be summarized in Fig. 1.

Schematically, digital cameras consist of a system of lenses, filters, a color filter array (CFA), an image sensor and a digital image processor (DIP).

Colour images may suffer from chromatic and spherical aberrations caused by the lenses, since there is no perfect lens. These effects are generally minimized by the combination of concave and convex lenses. Normally, there are auxiliary systems such as an auto-exposure control, an auto-focus control and an image stabilization unit to correct defects.

After passing through the lenses, light runs through a set of filters. An infrared filter absorbs or reflects the light allowing only the visible spectrum to go to the next filter, blocking infrared radiation which can decrease the sharpness of the image. An anti-aliasing filter reduces the aliasing, i.e. the effect which causes different continuous signals to become indistinguishable when sampled digitally. This phenomenon occurs when it is required to render a high resolution signal into a lower resolution form.

After the filters we find the real core of the camera: the image sensor. It consists of an array of pixels. When light hits the photodiodes, each one generates an analogic signal proportional to the intensity of the light, which is converted to a digital signal to be processed by the DIP. Most of the cameras use charge coupled device (CCD) sensors, although in mobile devices the use of complementary metal oxide semiconductor (CMOS) sensors is more common. These photodiodes do not get the colour at all, they just hold the brightness of the light providing a monochromatic output. To produce a colour image, a CFA is set before the sensor, so that each diode gets the light intensity for a single colour. Most cameras use model green–red–green–blue (GRGB) from Bayer CFA pattern. Bayer filter sensor output is an array of red, green and blue pixels with various intensities. Since each pixel only stores one of the three colours, the full image is originated by the DIP using different interpolation algorithms (demosaicing). Besides interpolation, DIP also executes other secondary processes like white balancing, noise reduction, image sharpening, aperture correction and gamma correction to generate a good quality picture.

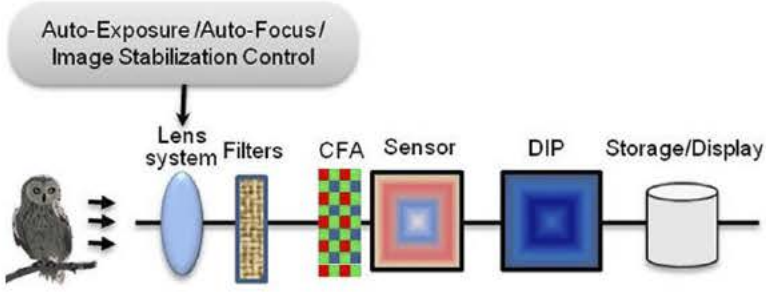


Fig. 1 Image acquisition process in digital cameras [4]

2.2 Image forensic analysis techniques

The area of image forensic analysis can be divided into two large branches: picture authentication and source identification [5].

With regards to the former, it determines if a picture was modified in any way since it was taken. The forensic algorithms used here must unfold certain features that could remain in the picture or verify the integrity of some properties that were introduced when the image was created [6–8].

The latter relates to the identification of the source of the image. It is based on the creation process of a certain device and the technology that was used. This kind of algorithms is based on statistical analysis of certain features which can be seen as "natural watermarks" belonging to the picture.

Apart from these two main lines, the information in the metadata which is inserted by devices when they take a picture should not be ignored. Supposing the information within the image is not altered in any way, each maker provides the forensic analyst with different but useful information such as GPS location, source, and technical features and so on. Focusing on the second main branch, it is necessary to take into account the device to be analyzed. In [4,9], a study is conducted regarding the potential elements which can be analyzed in mobile devices. Also in [10] how the process of image acquisition differs in mobile phones, DSCs and scanners is explained.

With regards to the set of techniques, those which use the content of the image itself are more robust than those based on metadata, although they are vulnerable as well [5].

According to [4] it is possible to classify the techniques into four categories:

2.2.1 Techniques based on lens aberration

During the process of acquisition, the lens produces aberrations on the image, such as spherical aberration, coma, astigmatism, field curvature, radial distortion and chromatic distortion. Of these, radial distortion is probably the worst.

Choi [11] proposes radial distortion as the best phenomenon to identify the source of the picture. Authors conclude that makers use different designs to compensate this effect, implying that each camera model would show a unique radial distortion pattern.

Van et al. [12] proposes lateral chromatic aberration as a way to identify the source of a photograph. Authors conclude that this technique is not suitable for identifying the source of different camera models of the same brand.

2.2.2 Techniques based on the use of sensor imperfections

They can be divided into two large branches: pixel defects or sensor noise patterns.

Geradts et al. [13] examines CCD pixel defects but it is not fully relevant in our case (CMOS). This technique includes point defects, hot points, dead pixel, pixel traps and cluster defects. The result noted that each one of the cameras had a different defect pattern. Nevertheless, it also noted that the number of defects in the pixels for a camera differed between pictures and varies greatly depending on the content of the image. It also revealed that the number of defects varied at different temperatures. Finally,

the study found that cameras with high-end CCD did not have this kind of problem, meaning that not all cameras suffered from this issue. It is also true that most cameras have additional mechanisms to compensate for this kind of problem.

Lukas et al. [14] proposes a method based on the non-uniformity of the pixels (pixel non-uniformity, PNU), which is a great source for the retrieval of noise patterns, which allows identifying the sensors and therefore the camera. The result for pictures with different sizes and cropped images is not satisfactory [4].

Costa et al. [15] proposes an approach for identifying the source of the picture considering open set recognition scenarios, under which we can't rely on the assumption of full access to all of the potential source cameras. This proposal consists of three stages: definition of regions of interest, determining the image features and the source camera identification. The use of regions of interest facilitates work with images of different resolutions. A total of 36 features based on sensor pattern noise to represent each image were obtained. In [16] new experiments with a bigger number of cameras and images were made. The results are good enough taking into account that the experiments were made in open set recognition scenarios.

2.2.3 Techniques based on the interpolation process of the CFA

In this category we find three main groups: traces of colour interpolation in colour bands, quadratic pixel correlation model and binary similarity measures. Regarding the first kind, Bayram et al. [17] researches the interpolation process in the CFA to determine the correlation structure used in each colour band, which can be used for classification purposes. The main supposition is that the interpolation algorithm and the CFA filters pattern design of each maker (even of each camera model) are somewhat different, which renders the correlation structures distinguishable. Using the expectation–maximization (EM) algorithm, two sets of features were obtained for the image classification: the image interpolation coefficients and the locations and magnitudes of the picks in the frequency spectre of probabilistic maps. The study determined that the percentage of success in the classification drops abruptly as the number of cameras used rises. Also, this technique is not very good for cameras of the same model and/or maker as they normally use identical CFA filters and similar interpolation algorithms. The study also states that it is not a good method for compressed pictures.

In [18] there is a proposal which suggests improvements to the one in [17], using additional features and obviously gathering better results. Even so, it is still limited for strongly compressed images.

In [19] we find another kind of technique. We get an array of coefficients of the quadratic pixel correlation model where the space periodic correlation among pixels is quadratic. Since similar cameras use the same demosaicing algorithm, this technique does not correctly distinguish between different cameras of the same maker. Moreover, it does not work well with compressed pictures.

In [20] the main supposition is that the proprietary CFA interpolation algorithms leave correlations along the bits of an image, which can be represented through this technique. 108 binary similarity measures were used as well as a set of 10 image

quality metrics. The results demonstrate how the success depends on the number of cameras used.

In [21] a technique for the identification of the source based on the information of the CFA interpolation process and a comparison between this and other techniques were presented. This technique shows three new sets of demosaicing features: weights, error cumulants (EC) and normalized group sizes (NGS). Since the number of features is too high a process called Eigenfeature regularization (ERE) was made to decrease the number.

The last technique to note is thoroughly examined in [22]. Four algorithms based on inter-channel correlation are proposed. These algorithms calculate variance maps (V-Maps) and classify them using a first nearest neighbor classification scheme (1NN). The authors conclude that the inter-channel correlation provides a new approach to the correlations between pixels introduced by the demosaicing process.

2.2.4 Techniques based on the image features

In [23] there is a group of features used to identify the source of a picture. The 34 selected features were divided into three categories: color features, image quality metrics and wavelet domain statistics. Just like the other methods, the success of this one was also dependent on the number of cameras selected.

In [24] a similar study was conducted for different sets of cameras. The success rate for cameras with similar CCD sensor is low. Thus, this method is inadequate to differentiate cameras of the same maker. Moreover, this technique requires the cameras to take pictures of the same content and resolution which is not always practical.

In [25] a genetic algorithm (GA) was used to automatically search for the best features and a support vector machine (SVM) classifier to identify the source of the camera by processing different pictures. These results are quite good and proved the high sturdiness of this technique compared to other kinds of image post-processing.

In [26] a set of experiments using the most common image features of wavelet, color, image quality metrics (IQM), statistical features of difference images and statistical features of prediction errors were used. In the experiments different combinations of varied types of features were proposed. A SVM were used for the classification. Authors concluded that some of the feature sets obtain a good average success rates for non-manipulated images but they were not so good for pictures with different types of alterations.

In [27] an image source identification technique using statistical models for *ridgelet* and *contourlet* subbands was shown. The experiments show that the features obtained by *contourlet* and *ridgelet* transforms achieve better results than those which are based on the wavelet transform. After features extraction an algorithm Sequential Floating Search (SFS) is used to select the features and a SVM for the classification. The *contourlets* and *ridgelets* are not only effective differentiating camera models, but also to differentiate between natural images or computer generated images, or to distinguish scanner images of the same manufacturer.

In [28] a method is proposed where the marginal density of the coefficients of the discrete cosine transform (DCT) in the low frequency coordinates and neighbouring joint density in DCT was used.

In [29] a method based on wavelets transform combined with sensor pattern noise is proposed. It extracts a set of 25 features which are used for a classification with a SVM. Several experiments were made with a different number of cameras and images. Authors concluded that the number of images used in the classification does not significantly affect the average success rate in this approach.

2.2.5 Techniques based on metadata

These are the simplest techniques and the aim of our research, although they strongly depend on the data the maker inserts as metadata when the picture is taken. Furthermore, this method is the most vulnerable to malicious changes by third parties. Nevertheless, assuming that it is possible to prove there is no kind of external modification, analyzing the large amount of metadata can greatly help the forensic analyst. Aside from this, metadata can also improve the results of forensic algorithms [30,31].

There are a huge amount of papers referencing the different types of metadata in pictures for search and classification purposes [24,32–34]. As stated before, these kinds of techniques, though simplest, depend on the metadata the maker may introduce. In fact, the most followed specification to identify the source of the camera, Exif [2], has two specific tags: "Make" and "Model". Unfortunately filling data in those tags is not mandatory.

Once the kind of technique or information to handle is chosen, an elaborate algorithm capable of identifying the source of a picture is required, with a high probability of success. There are several algorithms used for DSCs which must be studied to be adapted and used for mobile devices like [11–14,17–19,21,23,24,35–37].

In fact, there are currently algorithms and specific methods for mobile devices [15,16,22,25–29,38–40] which must be analyzed to minimize the likelihood of error.

A detailed comparison of different source identification techniques is presented in [41].

3 Metadata in images

Metadata is also known as "data about the data", i.e. relevant information complementing the main content of a digital document.

Each image format has different rules regarding how the different metadata formats are stored along with the file itself. Each one of these containers has its own format indicating the stored metadata properties, as well as their order and codification. In each container there is a separation by semantic criteria. These semantic groups are divided themselves into individual metadata properties. Each property has some specific data associated such as strings, numbers or arrays. Some properties, like image orientation, are not common to the different standard containers. However, others like copyright strings can be stored in several containers with similar information yet different semantic or slightly different structure.

The Exif specification is described below. It is the most common specification among digital cameras [34]. The Exif format defines a set of tagged image file format (TIFF) tags to describe photographic images. The specification uses existing file

Table 1 General scheme with markers of a JPEG file

SOI	Mark (1–n)				SOS			Image sata	EOI
FFD8	FF	Mark number (1 byte)	Data size (2 bytes)	Data (n bytes)	FFDA	Data size (2 bytes)	Data (n bytes)	Data (n bytes)	FFD9

formats such as joint photographic experts group (JPEG) [42] and TIFF Rev. 6.0 [43] and it adds specific tags to them. It is not supported in either JPEG 2000 or PNG.

There are several versions of the Exif specification. Each device supports a version which includes all previous ones. The Exif version used appears in an additional tag in the metadata. The last version of the specification is 2.3 from April 2010 [44].

Since the most used format in digital cameras and mobile devices is JPEG, the elements and data structures of JPEG/Exif are described here.

All JPEG files begin with the binary value 0xFFD8 (start of image, SOI) and end with 0xFFD9 (end of image, EOI). SOI and EOI are marks with no additional data unlike the other marks which have a fixed structure and associated data. In a mark, the "data size" field follows a byte alignment known as "Motorola" (*big-endian*). It is important to note that the data size includes the two bytes which indicate the length itself.

In JPEG format the mark 0xFFDA (start of stream, SOS), indicates the beginning of the image data itself, ending with EOI. Therefore, a general scheme with the possibility of n marks for a JPEG picture is shown in Table 1.

The mandatory markers in a JPEG/Exif file are: start of image (SOI), application marker segment 1 (APP1), define quantization table (DQT), define Huffman table (DHT), start of frame (SOF), start of stream (SOS) and end of image (EOI). It is also required the compressed data of the image itself.

Exif information is stored in segment APP1. There is a set of segments APPn not used by Exif, that can be used by makers to store any other kind of information while maintaining compatibility with Exif.

Exif uses the APP1 marker to avoid conflicts with the APP0 marker of the JPEG file interchange format (JFIF) format. After the size of the APP1 segment we find the value "Exif" in ASCII characters ('0x45786966') followed by 2 bytes '0x00' indicating the file follows Exif specification.

Exif uses the TIFF structure to store the data. Attributed information is recorded in tags specified in TIFF [43]. Attributed information specific to Exif is recorded using private tags reserved on TIFF for this standard. The private tags point to set this attribute information (Exif image file directory, IFD) which has 2 IFDs:

- 0th IFD: contains information about the image itself.
- 1st IFD: stores everything related to the thumbnail.

The thumbnail is used to preview the image without having to load and display the full resolution image. The thumbnail can be useful in an image forensic environment due to:

- The most significant parameters stored in the 1st IFD are: the compression details such as DQT and DHT, and dimensions of thumbnail.

- Not all software updates the thumbnail when saving the modified image.
- The process by which a thumbnail is created and stored is a bit different between camera manufacturers.

4 Binary analysis of images from mobile devices

Once the Exif specification has been introduced and as it is used in most of the mobile devices and DSCs [34], it has been deemed necessary to perform an analysis of several pictures taken with mobile devices. The goal is to get a deeper knowledge about the specification itself and check if it is closely followed by the makers.

Obviously, given the high number of tags Exif has, and since each image only has a subset of them, some structures and tags have been chosen for the analysis. This analysis has followed a logical order of structures from higher to lower levels (JPEG general structure, TIFF header, markers, IFD and specific tags).

The IFD consists of: a 2 byte count (number of fields), 12 byte field interoperability arrays, and 4 byte offset to the next IFD. Each of the 12 byte field interoperability consists of 4 elements:

- Tag: Each tag is assigned a unique 2-byte number to identify the field.
- Type: The following types are used in Exif standard: Byte (1), Ascii (2), Short (3), Long (4), Rational (5), Undefined (7), SLong (9) and SRational (10).
- Number of elements: The number of values in the IFD.
- Offset: This Tag records the offset from the start of the TIFF header to the position where the value itself is recorder. In cases where the value fits in 4 bytes, the values itself is recorded. If the value is smaller than 4 Bytes, the values is stored in the 4 Byte are starting from the left.

The structure of each IFD is shown in the Table 2.

For the first analysis we randomly selected two pictures taken from two phones (Samsung Galaxy S and Sony Ericsson W580i). These pictures had not been altered in any way.

Initially the files were checked to be JPEG. Through their general structure we see both files start with value 0xFFD8 (SOI) and end with 0xFFD9 (EOI). In the case of the Samsung Galaxy S we see the marker APP1 (0xFFE1), followed by its size 0x288E ("Motorola" alignment), i.e. 10882 bytes of data (including the 2 bytes of length). Thus, APP1 starts at 0x0004 and ends at 0x2892 (this byte not included). In the case of the Sony Ericsson W580i, we take the marker APP1 (0xFFE1), followed by the size 0x133D ("Motorola" alignment), i.e. 4925 bytes of data (including the 2 bytes of length). Thus, APP1 in this case starts at 0x0004 and ends in 0x1341 (this byte

Table 2 Structure of IFD tags

Bytes 0–1	Bytes 2–3	Bytes 4–7	Bytes 8–11
Tag	Type	Number of elements	Offset

Table 3 Analysis of *0th* IFD tags

Mobile phone	Entry 0th IFD			
	Bytes 0–1	Bytes 2–3	Bytes 4–7	Bytes 8–11
Samsung Galaxy S	0E01	0200	14000000	9E000000
	Image description	ASCII	20	158 bytes
	0F01	0200	14000000	B2000000
	Make	ASCII	20	178 bytes
Sony Ericsson W580i	0F01	0200	0E000000	86000000
	Make	ASCII	13	134 bytes
	1001	0200	06000000	A6000000
	Model	ASCII	6	166 bytes

not included). If we extract the APP1 marker for both pictures, we can see different results:

- Samsung Galaxy S: Next marker (address 0x2892) is 0xFFDB, which corresponds to DQT.
- Sony Ericsson W580i: Next marker (address 0x1314) is 0xFFC4, which corresponds to DHT.

With this data we see that after APP1, different images have different markers, which is allowed by Exif. The TIFF header data is stored inside the APP1 marker structure, where we can see both follow Exif, have "Intel" alignment and offset 0x00000008 to the first IFD.

Once some markers have been analyzed, we go to the next level, the IFDs. In the image from the Samsung Galaxy S we will analyze the structure of its first IFD and the two first tags. Following the TIFF header we find the bytes 0x0C00. As we use "Intel" alignment these bytes indicate how many entries the current directory has, in this case, the *0th* IFD. Thus, the *0th* IFD has 12 entries: The first entry of the directory 0x0E010200140000009E000000 is detailed in Table 3.

Therefore, to get the value of the tag "Image Description" we have to follow the offset since its size is greater than 4 bytes. As the length is 0x9E from the beginning of the TIFF header, the tag data starts at address 0xAA. From that byte we take 20 ASCII elements (7-bit ASCII) so the value is "SAMSUNG (12 blanks, 0x00)", ending in NULL (0x00) as indicated in Exif.

Looking at the next tag in directory *0th* IFD for the same file we find the second entry is 0x0F01020014000000B2000000, which is detailed in Table 3.

Therefore, to obtain the value of the tag "Make" we follow the offset as the length is 0xB2 from the TIFF header, the tag value starts at address 0xBE. From that byte we take 20 ASCII elements so the value of the tag is "SAMSUNG (12 blanks, 0x00)" ending in NULL (0x00) as stated in Exif. As we can see in Table 3 two different tags "Image Description" and "Make" can have the same value in an image, but their information has to be duplicated to comply with Exif.

Now we will analyze the IFD elements for the Sony Ericsson W580i picture. After the TIFF header we find bytes 0x0A00. Since it is using "II" ("Intel") the bytes 0x0A00 indicate how many entries the current directory has, in this case *0th IFD*. Thus, *0th IFD* has 10 entries. The first entry 0x0F0102000E00000086000000 is explained in Table 3.

Therefore, to get the value of the tag "Make" we have to follow the offset. As the length is 0x86 from the TIFF header, the tag value starts at address 0x92. From that byte we take 13 ASCII elements so the value of the tag is "Sony Ericsson0x00" ending in NULL (0x00) as the specification indicates. Looking up the next tag in the directory *0th IFD* we find the next entry 0x1001020006000000A6000000, explained in Table 3.

Therefore, to get the value of the tag "Model" we have to use the offset. As the length is 0xA6 from the TIFF header, the tag data starts at address 0xB2. From that byte we take 6 ASCII elements, resulting in the value "W580i0x00" ending in NULL (0x00), according to Exif specification.

4.1 Anomalies following Exif specification

After performing a similar analysis over thousands of pictures, there have been cases in which the specification was not follow utterly, even when the header stated the opposite. These are examples where the maker ensures complying the specs and actually not following them: In a picture taken with a Samsung Galaxy S, we detect an entry in directory IFD0 as 0x1001020008000000C6000000, explained in Table 4.

Therefore, according to the offset 0xC6 from the beginning of the TIFF header it points to address 0xD2, where we find tag "Model" being "GT-I9000" and a length of 8 as the header stated. It seems right at first, but to be accurate, this image does not comply 100 % with the Exif 2.2, since it claims to be type 2 (ASCII ending in NULL—0x00) and it is not. To store "GT-I9000" 9 elements are required (8 ASCII characters + 1 NULL) rather than 8 as stated by the directory entry.

This happens again in the Nokia N70, declared to be compliant with Exif 2.2. The tags analyzed are: 0xA004 ("Related Audio File") and 0xA420 ("Unique Image ID").

Tag entry "Related Audio File" is 0x04A0020001000003100520 2, whose interpretation is shown in Table 4.

Table 4 Tags *0th IFD* with anomalies

Mobile phone	Entry 0th IFD			
	Bytes 0–1	Bytes 2–3	Bytes 4–7	Bytes 8–11
Samsung Galaxy S	1001	0200	08000000	C6000000
	Model	ASCII	8	198 bytes
	04A0	0200	01000000	31005202
Nokia N70	Related Audio File	ASCII	1	0x31005202
	20A4	0200	01000000	31909504
	Unique image ID	ASCII	1	0x31909504

According to Exif specification, tag "Related Audio File" is of type ASCII and has 13 elements, 0x0000000D, but as shown in Table 4, it actually stores 0x00000001, i.e. 1 element, clearly violating the specification in at least two ways. Firstly, the tag "Related Audio File" fixes the data size to 13 bytes. Secondly, the specification sets the minimum data size to 4 bytes.

Once detected the file does not comply with the specification, the store data is 0x31005202. This is 1 in ASCII, followed by null (0x00), R in ASCII and the value 0x02 (STX in ASCII). This can cause problems for programs that extract Exif information due to the incoherence. Since these kinds of cases can be numerous, Exif information interpreters have to agree on a single way to treat this and similar cases. The options are as follows:

1. In the case of specification violation, the data is not shown, indicating a parsing error. This option is the most restrictive since it does not allow any ambiguity.

The following options show alternatives which allow the extraction of the information at the expense of ignoring the strict compliance of the Exif specification.

2. Extract all ASCII type data until the first null (0x00) is found. This option could generate errors because if ASCII strings do not end in null, unrelated data could be shown. In the worst case it could provoke memory overflow if the null character is not found in any tag.
3. Extract all data knowing only the size of them. This is the least restrictive option since it would show the ASCII characters even if they did not comply with the Exif specification.
4. Mix option between 2 and 3. I.e. extract all data knowing the size of each part and separating them using the null (0x00) as a separator.
5. Extract all tag data ignoring the size. Ie, if the size is less than or equal to 4 bytes, extract the next four bytes, and if greater than 4 bytes obtain the number of bytes indicated in the size from the corresponding offset. For treatment of null (0x00) we must choose between various options: (a) as blanks, (b) ignore (character empty) or (c) replaced by a special character outside the ASCII range valid for Exif (0–127).

Independent of the way used to show the Exif data, there is a problem in the creation of the file by the makers as they do not comply with the specification. Therefore, the option taken (1 or 2) could have important forensic consequences as this kind of anomalies could be abused by anti-forensic tools.

Another case of anomaly occurs in the same phone (Nokia N70) and in tag "Unique Image ID". Specification states it is of type ASCII and with 33 elements. According to the Exif specification, tag "Unique Image ID" is ASCII and has 33 elements, i.e.0x00000021, but as shown in Table 4 it only stores 0x00000001, that is, one element, clearly violating the specification, just like in the last case. Once it has been checked that this file does not comply with the specification, the tag stored is 0x31909504, taking into account 4 bytes, since the analysis considers that the fifth byte is the beginning of another tag. This fact violates again the specification, since ASCII must end in null (0x00) and it does not appear in the string. Also, there is another specification violation since ASCII characters use 7 bits (range 0–127, 0x00–0x7F), so characters 0x90 and 0x95 are fully out of the specification.

Here we showed some examples of anomalies detected after manual analysis, but others were found in other pictures and tags such as "Exif version", "Meetering Mode", "Exposure Program", "DateTimeOriginal",…

Therefore, we can conclude that many makers do not follow Exif specifications even when stating the opposite in the file, making it prone to problems when extracting the metadata through applications, as well as interoperability issues between devices. A detailed Analysis of the anomalies detected in metadatas are presented in [45].

5 Theia: tool for the forensic analysis of mobile devices pictures

It is obvious that the Exif metadata retrieval using the binary analysis manually is tedious and slow. Therefore, tools are needed for automatic extraction and graphical visualization in a user friendly way.

Theia, the developed application, carries out two levels of picture analysis: individual and in groups. The former obtains the Exif information of a single picture, finds alterations after comparing it to the thumbnail and automatically places it in Google Maps and Google Earth (if it has geopositioning info), as shown in Figs. 2 and 3.

The latter does a picture analysis as a group. Each group is utterly independent. The different analysis which can be executed over each group are: picture administration (adding or removing pictures), preset queries, modification analysis based on the stored thumbnail, advanced queries and image geopositioning.

- *Preset queries* Allows the creation of queries using Exif tags (and others that the application adds to help the forensic analysis) over the images of a selected group.

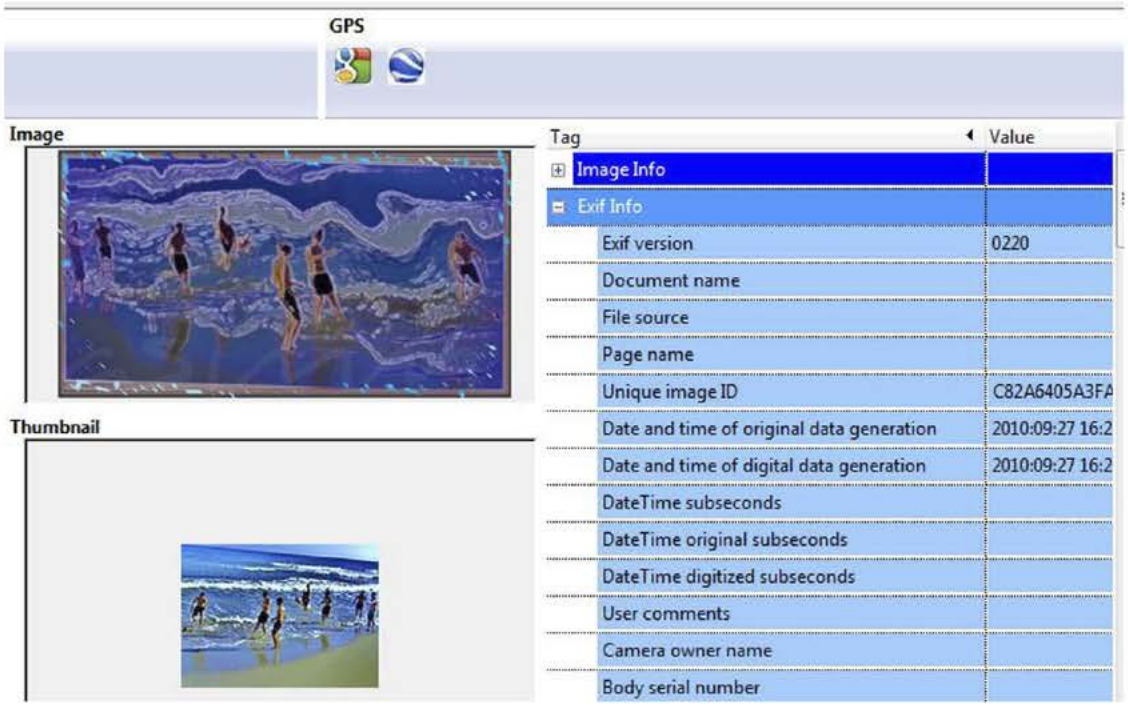


Fig. 2 Individual photo analysis

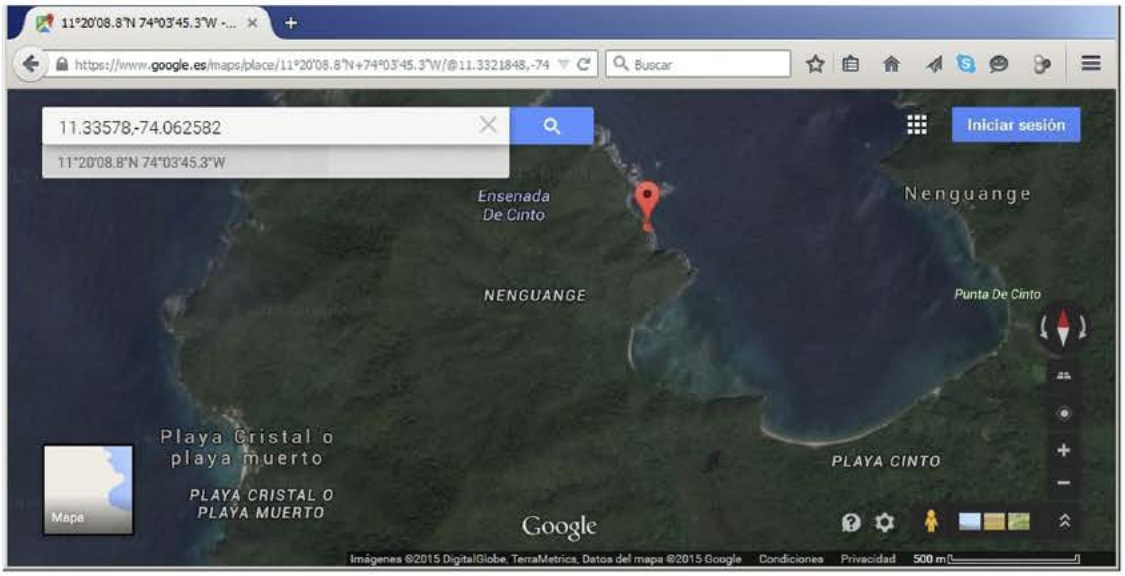


Fig. 3 Google Maps geopositioning of a photo

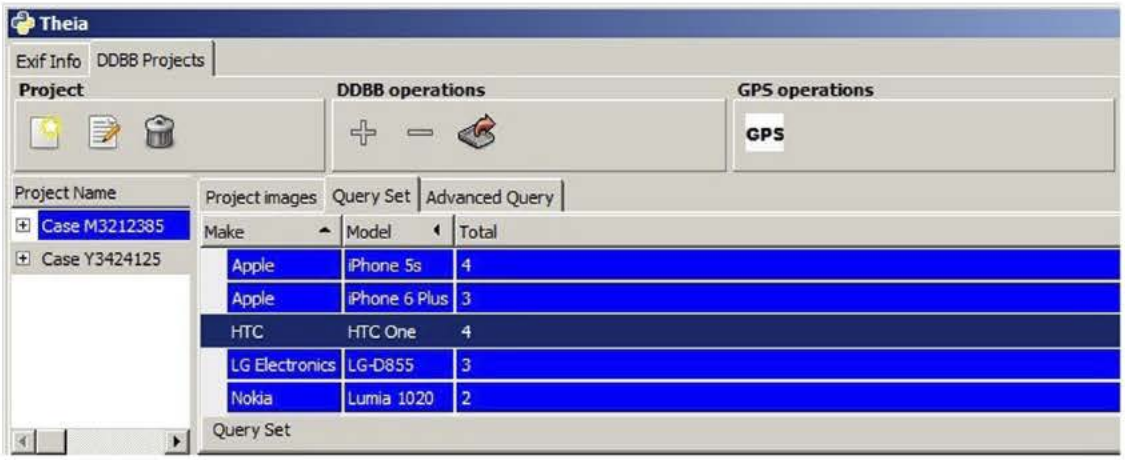


Fig. 4 Aggregation field selection

The query groups the pictures by the selected criteria and shows the number of pictures in each group, as shown in Fig. 4.

- *Analysis of modifications based on the stored thumbnail* It deems if modifications were done on the images of the group after they were taken. This operation occurs by the analysis of thumbnail stored on Exif metadata of image. A thumbnail is typically no larger in size than a few hundred square pixels, generally on the order of 160 × 120 pixels in resolution. This resolution varies across different camera manufacturers. The creation process of a thumbnail includes five basic steps: crop, filtering operations (blur and sharpen), contrast adjustment and JPEG compression. The specifics of these steps depend of camera manufacturers. This operation occurs by calculating the root mean square of the comparison of the thumbnail in the Exif info with the thumbnail generated from the analyzed image, and classifying the pictures into the following categories: without modifications, possibly modified and with modifications, represented in green, orange and red respectively, as shown in Fig. 5.

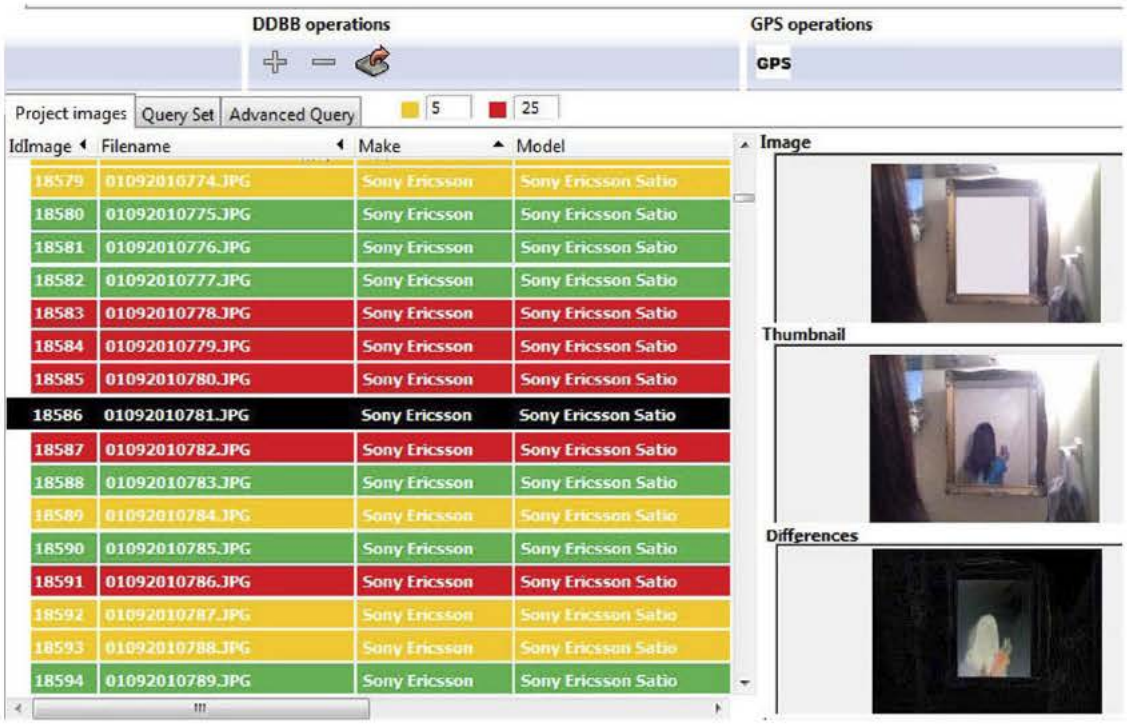


Fig. 5 Thumbnail analysis

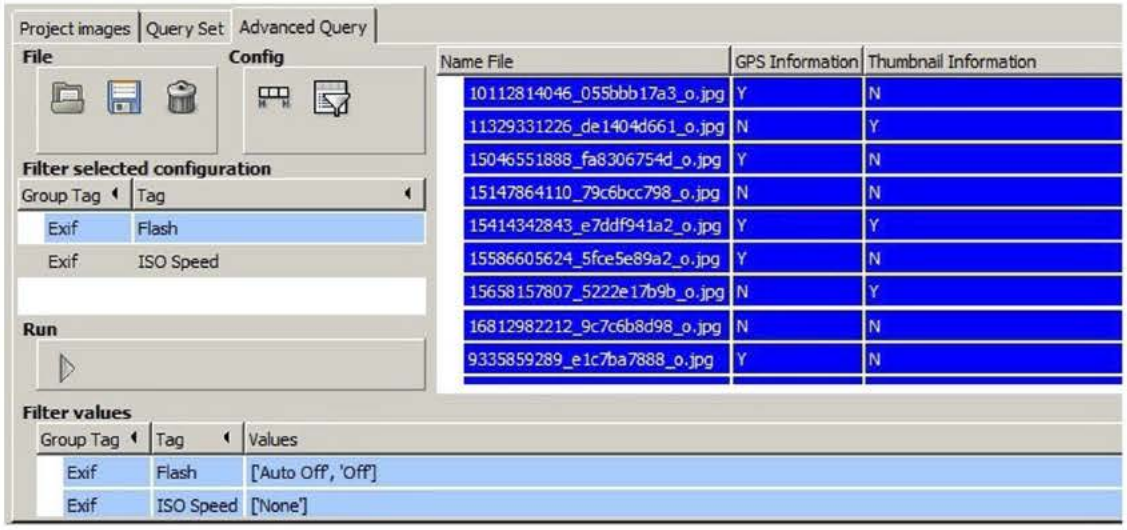


Fig. 6 Advanced query

- *Advanced queries* Supports the queries over pictures within a group configuring the Exif data to show and the filters to use. That is, it shows the information about the pictures of the selected fields matching the values of the configured filters. Furthermore, queries can be stored. A general vision is shown in Fig. 6.
- *Geopositioning* Analogous to the individual image processing there is a feature that allows the treatment of geoinformation for a group of images. This option grants the selection of some or all the pictures with geopositioning info for the creation of a map which places them in Google Maps. In the map, pictures are grouped by zones, and as the zoom increases, the coordinates are detailed. Figure 7

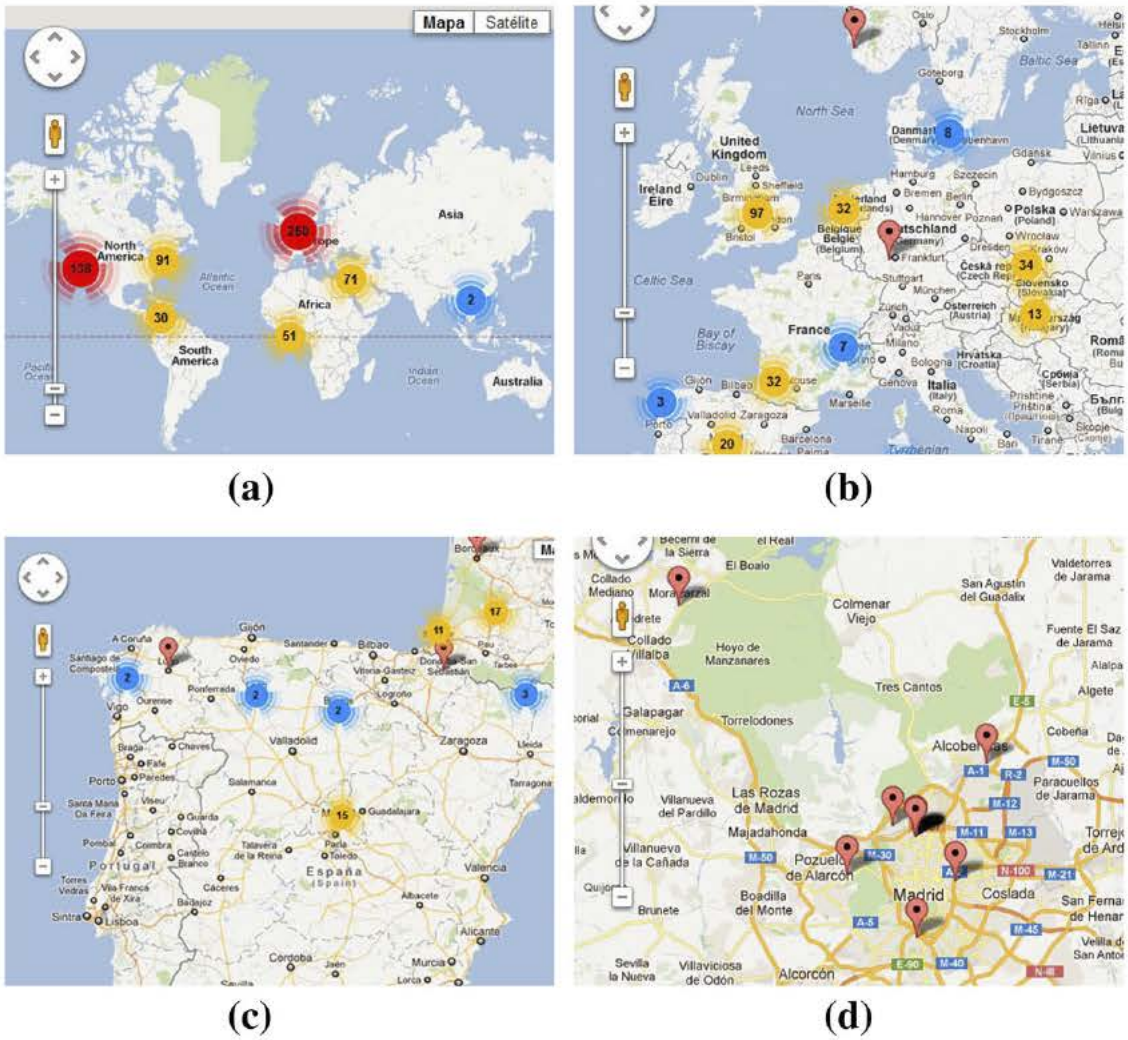


Fig. 7 Google Maps geopositioning of a group of images. (a) Zoom I (b) Zoom II (c) Zoom III (d) Zoom IV

shows an example of the generated map and the process of increasing the zoom in a concrete area (from Fig. 7a, d).

5.1 Comparison with other tools

To properly establish a comparison of *Theia* with others, extraction tools and Exif metadata editors for JPEG files were taken into account, but it did not exclude other related tools. The tools selected are *PhotoInfoEx*, *JHead*, *ExifTool*, *Exif Viewer* and *ExifPro Image Viewer*:

- *PhotoInfoEx* Digital photography program that allows editing or modification of certain Exif metadata or International Press Telecommunications Council (IPTC) [46] from the files in format JPEG and TIFF, and even some RAW files. The main advantages over *Theia* are better browsing over the files to examine, the export of metadata to Microsoft Excel and Microsoft Word and their printing.

As for disadvantages, there are problems extracting the Exif metadata that may not be 100 % accurate to the specification. As an example, there were pictures with wrong data in the tag "DateTimeOriginal". *PhotoInfoEx*, despite displaying an error or the string as it is stored, it formats the data internally and shows different data to those of the image, apparently right when they are not. Moreover, if does not allow for group analysis, with this being a key aspect, and finally it does not show any information regarding the thumbnail, so there is no way of knowing if there were alterations on the original image.

- *JHead* Powerful command line tool that extracts and edits Exif information from JPEG files [47]. Its only advantage is that it allows the extraction of IPTC and extensible metadata platform (XMP) metadata [48], even though they are not used by mobile devices. The main drawback is that, like PhotoInfoEx, it does not support group analysis. The lack of graphical interface makes it difficult to use and it has no geopositioning functionality in Google Maps or Google Earth and does not process thumbnails.
- *ExifTool* This application allows for the extraction and editing of metadata in a wide range of file formats [49] such as Exif, IPTC, XMP, JFIF. Furthermore, it can decode maker information (maker note) from lots of cameras. The pros and cons are the same as those of JHead and so are the conclusions drawn.
- *Exif Viewer* Firefox browser add-on that can extract Exif, IPTC and XMP metadata from JPEG files both local and remote [50]. The main advantage is that it is easy and quick to install, and to use, quite heavily limited. It allows the geopositioning in Google Maps, Google Earth, Yahoo! Maps and MSN Maps & Directions. The main drawback, as in previous cases, is the lack of group analysis. Furthermore, the way to present the results is poor and not user friendly. Therefore, it is no match for *Theia*.
- *ExifPro Image Viewer* Shows a limited amount of tags of Exif information of JPEG files [51]. The main advantage over all the other tools, including *Theia*, is the file browser. It offers a huge amount of possibilities to show, group and sort the pictures into directories. It is also the most powerful to show individual pictures. However, regarding extraction and manipulation of metadata, it is the most deprived tool. It just extracts about twenty Exif tags, which are not clearly presented. It has no geopositioning processing. It does not support group analysis and does not process the thumbnail. It has an option to export the Exif info from JPEG files to a text file. This facilitates the subsequent conversion of the information format to databases or spreadsheets. Exported data support group analysis, but the application does not support it directly, and requires the analyst to have advanced computer knowledge. Therefore, we conclude that this tool is for visualization and classification of pictures rather than metadata edition. The set of Exif metadata is very limited and not enough for forensic analysis.

Once *Theia* is compared to other similar tools, it is possible to conclude that as no one overmatches it, *Theia* is the one with higher functionality, power and versatility. Table 5 shows a comparison table of all examined tools.

None of them have group analysis or better Exif metadata extraction. *Theia* does not have the visualization of picture galleries as a main goal, but automates the tasks

Table 5 Comparison table among existing applications

Tool	Platform	Interface	EXIF Version	EXIF Data Visualization Edition[a]	EXIF Information	Metadata Format	Open Source	Free Software	Thumbnail Analysis	Observations
Theia	Windows	Graphical	2.3	Organized by IFD	No	Exif	Yes	Yes	Yes	Friendly and intuitive interface
	Mac OS									Group analysis
	GNU/Linux									
PhotoInfoEx	Windows	Graphical	2.21	Organized by IFD	Yes	Exif, IPTC	No	No	No	Metadata export to other formats
Jhead	Windows	Command line	Not specified	Not organized	Partial	Exif, IPTC, XMP	Yes	Yes	No	Difficult to use
	Mac Os-X									Extracts thumbnail but does not verify changes
	GNU/Linux									Reduced GPS functionalities
ExifTool	Windows	Command line	Not specified	Not organized	Yes	Exif, IPTC, XMP, JFIF	Yes	No	No	Difficult to use
	Mac OS X									Reduced GPS functionalities
	GNU/Linux									
Exif	Firefox plugin	Graphical	Not specified	Organized by IFD	No	Exif, IPTC, XMP	Yes	Yes	No	Remote image metadata
Exif Viewer										
ExifPro Image Viewer	Windows	Graphical	Not specified	Not organized	No	Exif	No	No	No	Limited Exif information tags
										No GPS functionality

[a] From a forensic point of view, it is sounder to avoid modifying images accidentally

of a forensic analyst regarding metadata as much as possible. This occurs more clearly in *Theia*.

6 Analysis of a set of pictures using Theia

In order to evaluate *Theia* as image forensic tool a number of tests have been configured with two types of image datasets from mobile devices using the functionalities described before. The goal of these tests is to search of useful data, patterns, image authentication or statistics about the Exif metadata.

6.1 Pictures dataset

Here are descriptions of two digital images datasets acquired from different source:

- *Personal image collection* These pictures have been taken from known people trying to get as much heterogeneity as possible regarding makers and models, as well as striving to gather as many pictures as we could of each one. The final collection is formed by 3751 pictures of 10 makes and 74 models. Table 6 show models grouped by maker with its corresponding number of pictures.
- *Internet downloaded images* These images were downloaded from Flickr.com (the popular photo-sharing website). Since we are interested in original images from mobile devices, it was necessary to eliminate any images that had been edited or altered by photo-editing software. First, only images tagged as "original" by Flickr were downloaded. Later, images with the following features were filtered: same models that personal image collection, 3-channel color JPEG, no duplicate images (by MD5 hashes comparing), no metadatas tags has been removed, the tags "modification" and "original" datetime inconsistent and resolution native to the camera of each mobile device. The final collection is formed by 2019 pictures of 10 makers and 74 models.

Unlike studies in the references cited, the number of camera models that were used in the analysis is much greater.

6.2 Analysis of the maker and model information

Since one of *Theia* targets is to identify the source of the image, "Query Set" is used to obtain the number of images grouped by maker and model. This analysis was performed using "Personal Image Collection" dataset. The data is checked in Table 6 to rate what makers do about the inclusion of these two metadata. Results are shown in Table 7.

As we can see in Table 7 some "IPhone 4" pictures have "IPhone" as model value, rendering them indistinguishable from those created by the "IPhone" (not IPhone 4), due to an update of the software. Similarly we can see that Nokia 6085 stores value "Nokia 0001" as a model. In Nokia N95 66 pictures are stored with value "N95" and 65 with "N95 8GB", since they are two versions of the same model.

Table 6 Mobile phones classified by maker and model from "Personal Image Collection" dataset

Make	Model	Total
Apple	Ipad 2 (A1)	19
	iPhone (A2)	24
	iPhone 3G (A3)	51
	iPhone 3GS (A4)	82
	iPhone 4G (A5)	49
HP	iPAQ hw6515 (B1)	35
	IPAQ RX3000 (B2)	35
HTC	8900 (C1)	38
	Desire (C2)	41
	Desire HD (C3)	162
	droid Incredible (C4)	32
	Droid Incredible 2 (C5)	27
	Evo 4G (C6)	38
	Hero (C7)	59
	MyTouch 4G (C8)	41
	TyTN ii (C9)	59
	Vodaphone HTCMagic (C10)	40
LG	CU720 (D1)	31
	KF750 (D2)	15
	KU990 (D3)	30
	Ku990i (D4)	144
	Rumor (D5)	26
	VX-8550 (D6)	13
	VX9700 (D7)	30
Nokia	5230 (E1)	19
	5300 (E2)	100
	5530 (E3)	31
	5800 (E4)	28
	6020 (E5)	30
	6085 (E6)	9
	6110 Navigator (E7)	35
	6120 Clasic (E8)	20
	6210 Navigator (E9)	29
	6230i (E10)	21
	6300 (E11)	133
	6303 classic (E12)	35
	6600 (E13)	36
	E61i (E14)	36
	E71 (E15)	5
	N70 (E16)	16
	N8 (E17)	32

Table 6 continued

Make	Model	Total
	N95 (E18)	131
	N96 (E19)	52
	N97 (E20)	37
	N97 mini (E21)	54
Palm	Centro (F1)	28
	Pre (F2)	20
	Treo 680 (F3)	22
Research In Motion	BB 8100 (G1)	38
	BB 8300 (G2)	31
	BB 8320 (G3)	16
	BB 8330 (G4)	34
	BB 8520 (G5)	213
	BB 8900 (G6)	37
	BB 9000 (G7)	32
	BB 9550 (G8)	30
	BB 9630 (G9)	31
	BB 9800 (G10)	30
Samsung	Caliber (H1)	8
	Captative (H2)	24
	Galaxy 3 (H3)	33
	Galaxy S (H4)	15
	Galaxy S II (H5)	30
	H1 (H6)	6
	Omnia 7 (H7)	37
	Pixon (H8)	30
	SGH-F250L (H9)	4
	Start (H10)	39
	Wave (H11)	17
Sony Ericsson	C702 (I1)	79
	C905 (I2)	40
	K550i (I3)	13
	LT15i (I4)	11
	Satio (I5)	61
	T707 (I6)	102
	Vivaz (I7)	16
	W580i (I8)	158
	W705 (I9)	21
	W800i (I10)	39
	W910i (I11)	7
	X10 Mini (I12)	10
	Z610i (I13)	61

Table 6 continued

Make	Model	Total
Motorola	Atrix MB860 (J1)	35
	Backflip mb300 (J2)	47
	Cliq (J3)	30
	Defy mb525 (J4)	22
	Droid (J5)	31
	Droid X (J6)	79
	Droid X2 (J7)	54
	W377 (J8)	20

Table 7 Analysis results about maker and model information

Dataset model	Exif information		
	Make	Model	Total
Apple			
A1	Apple	iPad 2	19
A2	Apple	iPhone	24
A3	Apple	iPhone 3G	51
A4	Apple	iPhone 3GS	82
A5	Apple	iPhone 4	45
	Apple	iPhone	4
HP			
B1		HP iPAQ HW6515	35
B2	HP	iPAQ rx3000	35
HTC			
C1	HTC-8900	HTC-8900	38
C2	HTC	HTC Desire	41
C3	HTC	Desire HD	162
C4	HTC	ADR6300	32
C5	HTC	ADR6350	27
C6	HTC	PC36100	38
C7	HTC	HTC Hero	59
C8	HTC	myTouch 4G	41
C9	HTC	HTC_TyTN_II	59
C10	HTC	HTC Magic	38
	Vodafone	HTC Magic	2
LG			
D1	LG ELECTRONICS	CU720	31
D2	LG Electronics	KF750	15
D3	LG Electronics	KU990	30
D4	LG Electronics	KU990i	144
D5	LG Electronics	RUMOR	26
D6	LG Electronics	VX-8550	13
D7	LG Electronics	VX-9700	30

Table 7 continued

Dataset model	Exif information		
	Make	Model	Total
Nokia			
E1	Nokia	5230	19
E2	Nokia	5300	100
E3	Nokia	5530	31
E4	Nokia	5800 Xpres	28
E5			30
E6	Nokia	0001	9
E7	Nokia	6110	35
E8	Nokia	6120c	20
E9	Nokia	6210 Navig	29
E10			21
E11	Nokia	6300	133
E12	Nokia	6303 classic	35
E13	Nokia	6600i-1c	36
E14	Nokia	E61i	36
E15	Nokia	E71	5
E16	Nokia	N70-1	16
E17	Nokia	N8-00	32
E18	Nokia	N95	66
	Nokia	N95 8GB	65
E19	Nokia	N96	52
E20	Nokia	N97	37
E21	Nokia	N97 mini	54
Palm			
F1		Palm Centro	28
F2	Palm	Pre	20
F3		Treo 680	22
Research In Motion			
G1	RIM	BlackBerry 8100 Series	38
G2	Research In Motion	BlackBerry 8300	31
G3	Research In Motion	BlackBerry 8320	16
G4	Research In Motion	BlackBerry 8330	34
G5	Research In Motion	BlackBerry 8520	213
G6	Research In Motion	BlackBerry 8900	37
G7	Research In Motion	BlackBerry 9000	32
G8	Research In Motion	BlackBerry 9550	30
G9	Research In Motion	BlackBerry 9630	31
G10	Research In Motion	BlackBerry 9800	30

Table 7 continued

Dataset model	Exif information		
	Make	Model	Total
Samsung			
H1	SAMSUNG	SCH-R860	8
	SAMSUNG	SPH-M570	8
H2	SAMSUNG	SGH-I897	24
H3	SAMSUNG	GT-I5800	33
H4	SAMSUNG	GT-I9000	15
H5	Samsung	GT-I9100	30
H6	SAMSUNG	Vodafone 360 Samsung H1	6
H7	SAMSUNG	GT-I8700	37
H8	SAMSUNG	GT-M8800	30
H9	SAMSUNG	SGH-F250L	4
H10	Samsung	GT-S5230	39
H11	SAMSUNG	GT-S5333	17
Sony Ericsson			
I1	Sony Ericsson	C702	79
I2	Sony Ericsson	C905	40
I3	Sony Ericsson	K550i	13
I4	Sony Ericsson	LT15i	11
I5	Sony Ericsson	U1a	26
	Sony Ericsson	U1i	35
I6	Sony Ericsson	T707	102
I7	Sony Ericsson	U5i	16
I8	Sony Ericsson	W580i	158
I9	SONY ERICCSON	W705	21
I10	Sony Ericsson	W800i	39
I11	Sony Ericsson	W910i	7
I12	SEMC	X10a	10
I13	Sony Ericsson	Z610i	61
Motorola			
J1	Motorola	MB860	35
J2	Motorola	MB300	47
J3	Motorola	MB200	30
J4	Motorola	MB525	22
J5	Motorola	Droid	31
J6	Motorola	DROIDX	79
J7	Motorola	DROID X2	54
J8	Motorola	C261	20

With regards to Palm se we see that models "Treo 680" and "Centro" have NULL in tag "Make".

The first conclusion of this analysis is positive since it is possible to see a high rate of compliance by makers when they store maker and model values although in some cases, this may not be completely right.

Another important aspect of this analysis is the lack of uniformity of the makers when they store maker and model information in Exif tags, since they do not always use the same string for make and even sometimes they use the same string for different models (as in IPhone and IPhone4), which can be prone to mistakes during the identification. For instance, Sony Ericsson uses either "SEMC", "Sony Ericsson" or "SONY ERICCSON" depending on the model to store the make, and "Research In Motion" uses in "BlackBerry" models either the string "RIM" or "Research In Motion". In addition, mobiles designed by a specific mobile company like HTC Magic which have a model for Vodaphone store this name in the tag "Make", which is crucial information from a forensic point of view as it reveals sensitive data about the user.

We used "Advanced Query" to identify which pictures do not have maker and model information. The result shows those pictures, all from the "Nokia 6020" and "Nokia 6230".

This analysis was repeated using "Internet Downloaded Images" dataset. Later, the results of both analysis were compared. It was found that images with value empty in tag "Software", have the same characteristics to "Personal Image Collection" dataset. The similarities among them are:

- Tags enabled in both datasets are the same.
- The errors committed by the manufacturer to store some tags are the same in both datasets.
- Make and Model tags are similarly stored. However, this does not occur if the owner of the mobile personalizes this information from the device itself.

6.3 Image and Exif tags information

Analyzing the Exif tags in blocks "Image Info" and "Exif Info", we studied using "Query Set" those pictures with no information in any of those fields. This analysis was performed using "Personal Image Collection" dataset. The result of this is that all images have information in both fields.

Furthermore, in block "Image Info" field "Software Used" is analyzed using "Query Set". This field may be relevant for forensic analysis since it may reveal data such as the software used when creating the image. Results report uniformity in version naming for each maker. Notwithstanding this, there is strong disagreement among the different makers. Also, the software used varies among different mobile operators for a given model. For instance, in a Sony Ericsson W580i tag "Software Used" has values such as "R8BE001 prgCXC1123474_ORANGE_LA 0.0" (Orange) and "R8BE001 prgCXC1123362_GENERIC_L 0.0" (for any other operator).

Fig. 8 Display of photos in Google Maps

6.4 Analysis of the GPS information

Analyzing the Exif tags in block "GPS Info" using "Personal Image Collection" dataset, we examined the pictures with no information here. This analysis is subjective since it depends on whether the terminal had integrated GPS, the user had it on and allowed the insertion of GPS information when the picture was taken. Nevertheless, results state that 2918 of the checked pictures had no GPS information whilst 823 did have. Later, we studied pictures with GPS info but without latitude and longitude and 191 were obtained. Using "Advanced Query" 144 *LG KU990i* were detected, as well as 16 *Sony Ericsson Vivaz*, 30 *Motorola Droid x* and 1 *Sony Ericsson Satio*.

Regarding *LG KU990i*, its technical specifications indicate it has no GPS system, but even so it stores, in all images, a tag "GPSVersionID" with the value "Version 2.3", which is not mandatory without GPS information. *Sony Ericsson Vivaz* and *Sony Ericsson Satio* do have GPS (specifically A-GPS). *Sony Ericsson Vivaz* allows to geotag the pictures but does not store the corresponding GPS information, whilst Sony Ericsson Satio stores the tags "GPSVersionID" with value "0.0", "GPSAltitudeRef" with value "Sea level" and "GPSAltitude" with value "0".

This last case may be because the picture was taken with the A-GPS off, or the user did not allow the inclusion of GPS information in the image. Regardless, it makes no sense to fill in the three tags with apparently invalid data.

The 632 images with geopositioning data can be seen in Fig. 8.

6.5 Thumbnail information analysis

As is discussed above, the thumbnail is a copy of the original image but smaller in size. The thumbnail is stored in the IFD1 of the JPEG file. Having this thumbnail is a great step forward to forensic analysis, because the original image can be manipulated and we can obtain conclusions with the thumbnail analysis. The Exif specification

Table 8 "Thumbnail Info" analysis

Make	Model	Thumbnail		Total
		No	Yes	
Apple	ipad 2	13	6	19
	Iphone	5	19	24
	Iphone 3g	2	49	51
	iphone 3GS	19	63	82
	iphone 4g		49	49
Research In Motion	BB 8100	38		38
	BB 8300	31		31
	BB 8320	16		16
	BB 8330	34		34
	BB 8520	213		213
	BB 8900	8	29	37
	BB 9000	29	3	32
	BB 9550	30		30
	BB 9630	31		31
	BB 9800	30		30
HP	iPAQ hw6515	35		35
	iPAQ rx3000	35		35
HTC	8900	2	36	38
	desire	13	28	41
	desire hd	4	158	162
	droid incredible	14	18	32
	Droid Incredible 2	20	7	27
	evo 4g	11	27	38
	Hero		59	59
	myTouch 4G	5	36	41
	tyTN ii	28	31	59
	vodaphone htc magic	6	34	40
LG	CU720		31	31
	KF750	1	14	15
	KU990		30	30
	ku990i		144	144
	rumor	5	21	26
	VX-8550		13	13
	VX9700		30	30
Motorola	Atrix mb860	1	34	35
	backflip mb300	1	46	47
	Cliq	1	29	30
	Defy mb525	2	20	22
	droid		31	31
	droid x		79	79

Table 8 continued

Make	Model	Thumbnail		Total
		No	Yes	
	Droid x2		54	54
	W377	20		20
Palm	Centro		28	28
	Pre	4	16	20
	Treo 680		22	22
Nokia	5230		19	19
	5300	12	88	100
	5530		31	31
	5800		28	28
	6020	30		30
	6085	9		9
	6110 Navigator		35	35
	6120 Clasic		20	20
	6210 Navigator	2	27	29
	6230i	6	15	21
	6300	7	126	133
	6303 classic		35	35
	6600		36	36
	E61i		36	36
	e71		5	5
	N70		16	16
	N8		32	32
	N95	65	66	131
	N96	2	50	52
	N97	1	36	37
	N97 mini		54	54
Samsung	Galaxy S II		30	30
	Galaxy S		15	15
	Caliber	3	5	8
	Captative		24	24
	Galaxy 3	1	32	33
	H1		6	6
	Omnia 7		37	37
	Pixon		30	30
	SGH-F250L	4		4
	start		39	39
	Wave	2	15	17

Table 8 continued

Make	Model	Thumbnail		Total
		No	Yes	
Sony Ericsson	C702	11	68	79
	c905		40	40
	k550i	1	12	13
	LT15i		11	11
	Satio	1	60	61
	T707		102	102
	vivaz		16	16
	W580i		158	158
	W705		21	21
	w800i	4	35	39
	w910i	4	3	7
	X 10 Mini		10	10
	Z610i		61	61

allows the file to store a thumbnail image used to index the main image. The Exif 2.3 specification does not require images to have a thumbnail, but it does recommend it. This type of deficiency is common and it is found in many models. There are cameras and models that store the thumbnail in Maker Notes tag, however in the different analysis we have not found any photo to have this characteristic.

Then, two types of analysis with thumbnail stored in the Exif metadata of an image are performed: (1) Analysis of thumbnail information stored in Exif metadata in the block "Thumbnail Info". (2) Modification detection analysis on images performed after its capture.

6.5.1 Analysis of thumbnail information in "Thumbnail Info"

This analysis was performed using "Personal Image Collection" dataset. The Exif tags in "Thumbnail Info" were examined using "Query Set" to find out what images had no information regarding thumbnails. The result shown in Table 8 reveals that 2883 images (76.85 %) have thumbnail information and 868 (23.16 %) have not, of which 456 (52.53 %) belong to maker *Research In Motion*. Through this analysis it was proved that *Research In Motion* did not store a thumbnail in the Exif information in models before BB 8900. Furthermore, some specific models from makers such as Palm Centro, Treo 680, HP iPAQ hw6515, Motorola W377, Nokia 6020 and Nokia 6085 have no thumbnail at all. The use of thumbnails is recommended in Exif, so not having them, while not against the standard, is fairly unusual.

Another type of analysis to identify the compression used on the thumbnail should be noted. The thumbnail can be in a compressed or uncompressed format as long as the main image format is compressed (for instance, the JPEG format). In the case that

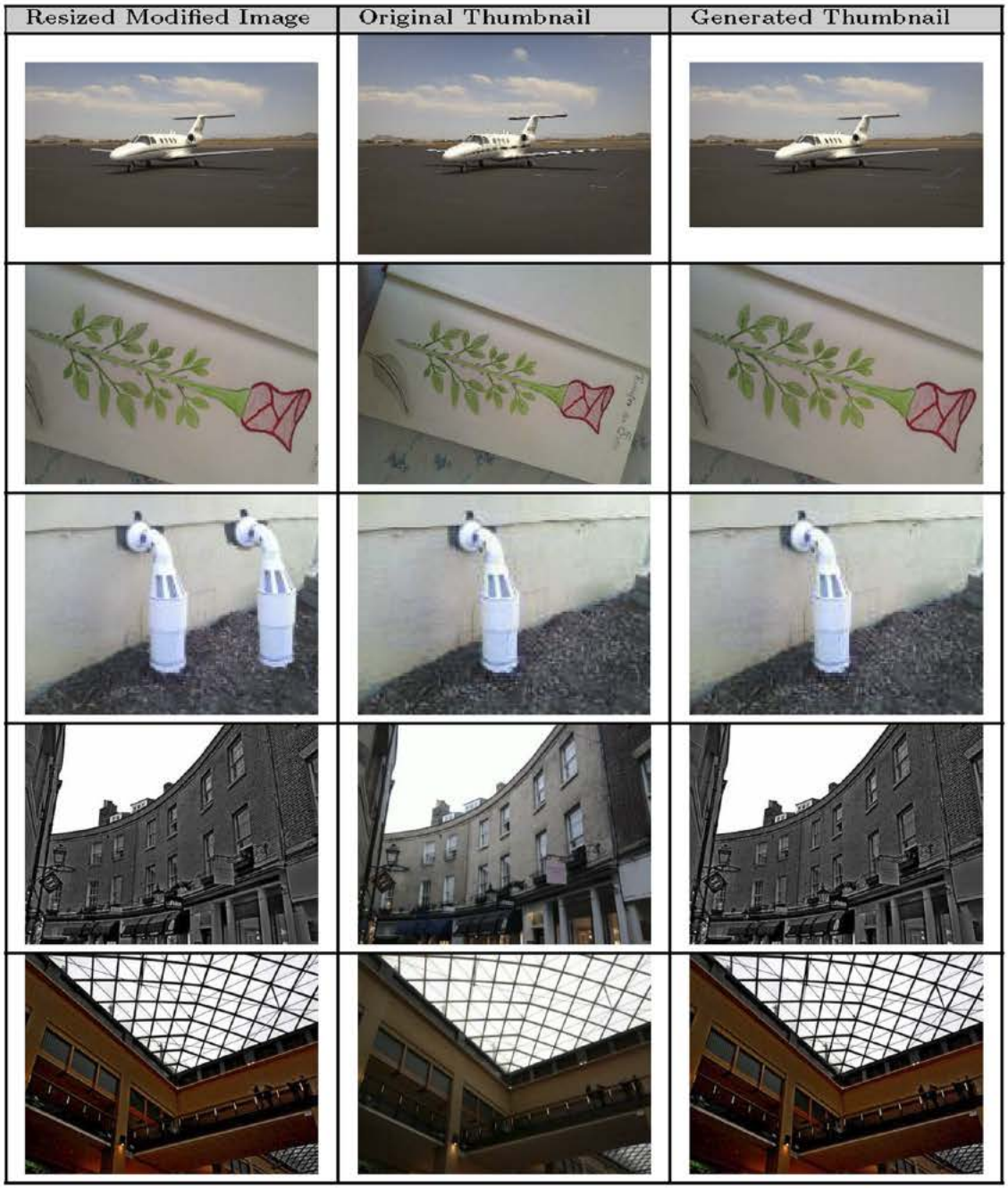


Fig. 9 Example of modified images

the main image is in an uncompressed format, the thumbnail can only be stored in an uncompressed format. However, if the image has compressed JPEG format, the thumbnail may not. The result of this shows that 100 % of thumbnails are in JPEG compressed format. This analysis was performed using "Personal Image Collection" and "Internet Downloaded Images" datasets.

6.5.2 Modification Detection Analysis

First, an analysis with several images to verify that the thumbnail can be used reliably for image authentication. 20 pictures of different mobile devices were selected from

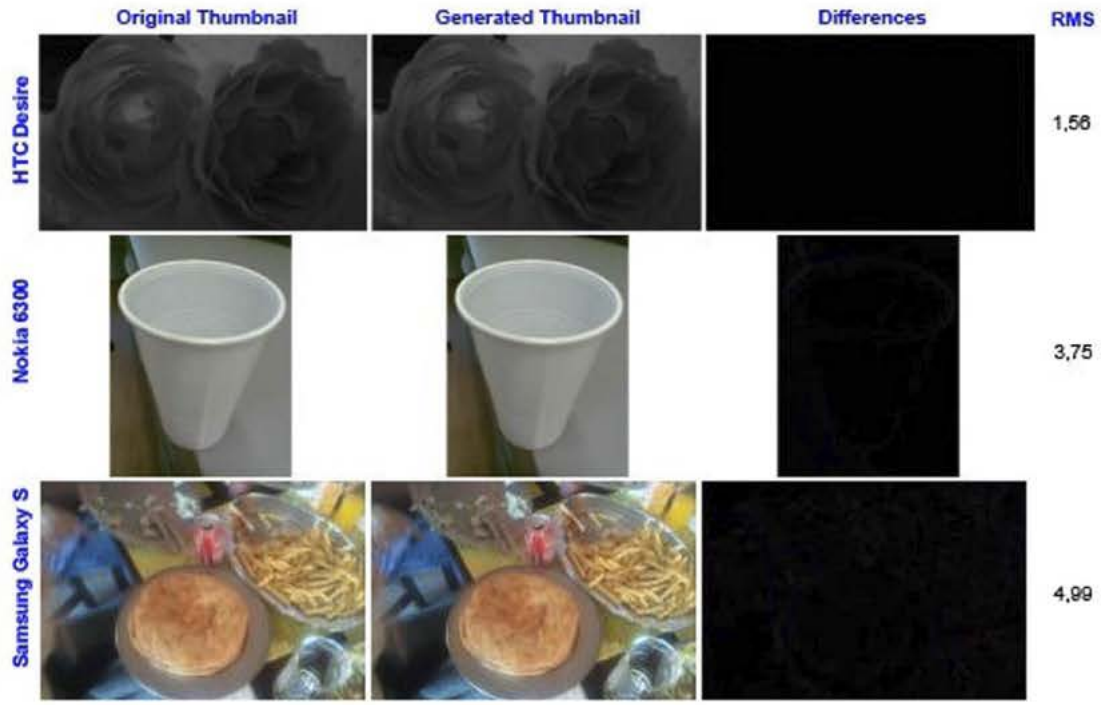


Fig. 10 Not modified images (RMS < 5)

Fig. 11 Possibly modified images (RMS between 5 and 25)

"Personal Image Collection" dataset. 10 of these images were modified with one of the following types of changes: retouching, composition and copy/move regions or objects. The tools used to edit pictures were Photoshop CS5 and Gimp. Then, thumbnail tags ("JpegIFOffset" and "JpegIFByteCount") and "Thumbnail Info" block were extracted from the Exif metadata of original images and, subsequently, were

Fig. 12 Modified images (RMS > 25)

replaced in the corresponding tags of the Exif metadata of the modified image. Finally, *Theia* was used to detect modified images among the 20 selected images.

Figure 9 shows some images that have been manipulated to perform this analysis. The Figure shows the resized modified image, the original thumbnail extracted from

Table 9 Analysis results of modified images

Classification	Observation	DatasetA	DatasetB
Without Thumbnail		872	0
	Rotated image	5	279
Not modified (RMS $<$ 5)	Different size	131	0
	Same size and orientation	186	683
	Rotated image	26	255
Possibly modified (5 $\leq$ RMS $\leq$ 25)	Different size	428	0
	Same size and orientation	1881	419
	Rotated image	2	157
Modified (RMS $>$ 25)	Different size	100	0
	Same size and orientation	120	226
Total		3751	2019

the Exif metadata and the generated thumbnail using *Theia* from the content of the modified image.

The results showed a 95 % accuracy in the modified images detection process. This analysis was based on the calculation of the Root Mean Square (RMS) and its resulting values were divided into three groups: "without modifications" for those images with RMS under 5, "possibly modified" for those with RMS between 5 and 25, and "with modifications" for those with RMS over 25. Figures 10, 11 and 12 show a group of images of each kind obtained through this analysis.

Lastly, an analysis of modification identification was made comparing the thumbnail in the Exif information to the thumbnail generated from the image. This analysis was performed using "Personal Image Collection" (A) and "Internet Downloaded Images" (B) datasets. The result of this analysis can be seen in Table 9. In the comparison process of the two thumbnails (generated and extracted from the Exif metadata) *Theia* check the parameters of both thumbnail according to the Exif 2.3 standard, and then, this information is added as an observation (this information is shown in column 2 of Table 9).

6.6 Analysis of the maker note information

Here the "Maker Note Info" Exif tags is reviewed, searching with "Query Set" those images without information of this kind. Results show that 0 % of them have "Maker Note Info" data. This implies makers currently do not insert any information, although this statement requires a deeper study in order to extrapolate it to all mobile devices.

6.7 Analysis of the interoperability information

After analyzing the Exif tags in the "Interoperability Info" block, using "Query Set" we found that 2082 images had this information, while 1669 did not.

7 Conclusions and future work

Concerning *Theia*, there have been differences shown with other tools regarding metadata extraction. Other applications allow a wider range of metadata formats. Nevertheless, most images in mobiles have a JPEG/Exif format, and thus *Theia* presents no great disadvantages regarding that aspect.

The main advantage we can spot is that *Theia* has to meet the needs of a forensic analyst as a main target. No other application has been found that can compete in this aspect since they cannot perform the following operations:

- Image analysis in groups, the revised tools only offer metadata information of individual images.
- Compare the thumbnail stored in the Exif metadata of each image in order to find modifications performed after the photos were taken.
- View in Google Maps as a group.

Regarding the studies we carried out comparing our tool with similar ones, it should be noted that they have been performed using our own dataset which is a sample of the entire universe of brands and models of current mobile devices rather heterogeneous and numerous.

We must never forget that all analysis performed over Exif metadata is unfortunately vulnerable to malicious modifications by third parties. Moreover, identification of the source of the image using the metadata totally depends on its insertion by the manufacturer. This creates the need for the application of more robust techniques to properly identify the source based on the content of the image itself rather than its metadata. Nonetheless, metadata gives useful information for the forensic analyst such as geopositioning, now impossible to deduce just by the contents of the image and the evidence of subsequent modifications after taking the photo, based on its thumbnail. We therefore conclude that it is necessary to add new techniques to identify the source of an image, but the functionalities which deal with Exif metadata are also deemed essential.

Acknowledgments The authors acknowledge to "Programa de Financiación de Grupos de Investigación UCM validados de la Universidad Complutense de Madrid - Banco Santander".

References

1. Hsu J (2010) The worldwide mobile phone camera module market and Taiwan's industry. Market Intelligence & Consulting Institute (MIC), pp 1–18
2. Baer R (2010) Resolution limits in digital photography: the looming end of the pixel wars—OSA technical digest (CD). In: Proceedings of the imaging systems, p. ITuB3. doi:10.1364/IS.2010.ITuB3
3. Srivastava L (2005) Mobile phones and the evolution of social behavior. Behav Inf Technol 24(2):111–129
4. Van Lanh T, Chong KS, Emmanuel S, Kankanhalli MS (2007) A survey on digital camera image forensic methods. In: Proceedings of the IEEE international conference on multimedia and expo, pp 16–19. doi:10.1109/ICME.2007.4284575
5. Gloe T, Kirchner M, Winkler A, Bohme R (2007) Can we trust digital image forensics? In: Proceedings of the 15th international conference on multimedia, pp 78–86. doi:10.1145/1291233.1291252
6. Chang C (2002) A steganographic method based upon JPEG and quantization table modification. Inf Sci 141(1–2):123–138. doi:10.1016/S0020-0255(01)00194-3

7. Chamlawi R, Khan A (2010) Digital image authentication and recovery: employing integer transform based information embedding and extraction. Inf Sci 180(24):4909–4928. doi:10.1016/j.ins.2010.08.039
8. Tsai HH (2007) Color image watermark extraction based on support vector machines. Inf Sci 177(2):550–569. doi:10.1016/j.ins.2006.05.002
9. Thing VLL, Ng KY, Chang EC (2010) Live memory forensics of mobile phones. Digit Investig 7:S74–S82. doi:10.1016/j.diin.2010.05.010
10. Mckay C, Swaminathan A, Gou H, Wu M (2008) Image acquisition forensics: forensic analysis to identify imaging source. In: Proceedings of the IEEE international conference on acoustics, speech and signal processing (ICASSP), pp 1657–1660. doi:10.1109/ICASSP.2008.4517945
11. Choi KS (2006) Source camera identification using footprints from lens aberration. In: Proceedings on digital photography II. SPIE—The International Society for Optical Engineering, vol 852, no 6069, p 60, 690J–60, 690J–8. doi:10.1117/12.649775
12. Van LT, Emmanuel S, Kankanhalli M (2007) Identifying source cell phone using chromatic aberration. In: IEEE international conference on multimedia and expo, pp 883–886. doi:10.1109/ICME.2007.4284792
13. Geradts ZJ, Bijhold J, Kieft M, Kurosawa K, Kuroki K, Saitoh N (2001) Methods for identification of images acquired with digital cameras. In: Proceedings of the international conference on enabling technologies for law enforcement and security, vol 4232, pp 505–512. doi:10.1117/12.417569
14. Lukas J, Fridrich J, Goljan M (2006) Digital camera identification from sensor pattern noise. IEEE Trans Inf Foren Secur 1(2):205–214. doi:10.1109/TIFS.2006.873602
15. Costa FDO, Eckmann M, Scheirer WJ, Rocha A (2012) Open set source camera attribution. In: Proceedings of the 25th conference on graphics, patterns and images. IEEE, New York, pp 71–78. doi:10.1109/SIBGRAPI.2012.19
16. de Costa OF, Silva E, Eckmann M, Scheirer WJ, Rocha A (2014) Open set source camera attribution and device linking. Pattern Recognit Lett 39(0):92–101
17. Bayram S, Sencar H, Memon N, Avcibas I (2005) Source camera identification based on CFA interpolation. In: Proceedings of the IEEE international conference on image processing, vol 3, pp 69–72. doi:10.1109/ICIP.2005.1530330
18. Bayram S, Sencar HT, Memon N (2006) Improvements on source camera-model identification based on CFA interpolation. In: Proceedings of the working group international conference on digital forensics. Springer, Berlin, pp 24–27
19. Avcibas I, Memon N, Sankur B (2003) Steganalysis using image quality metrics. IEEE Trans Image Process 12(2):221–229. doi:10.1109/TIP.2002.807363
20. Celiktutan O, Avcibas I, Sankur B, Ayerden NP, Capar C (2006) Source cell-phone identification. In: Proceedings of the IEEE 14th signal processing and communications applications, pp 1–3. doi:10.1109/SIU.2006.1659882
21. Cao H, Kot AC (2010) Mobile camera identification using demosaicing features. In: Proceedings of 2010 IEEE international symposium on circuits and systems (ISCAS). IEEE, New York, pp 1683–1686
22. Ho J, Au O, Zhou J, Guo Y (2010) Inter-channel demosaicking traces for digital image forensics. In: 2010 IEEE international conference on multimedia and expo (ICME), pp 1475–1480. doi:10.1109/ICME.2010.5582951
23. Mehdi KL, Sencar HT, Memon N (2004) Blind source camera identification. In: Proceedings of the international conference on image processing, vol 1. IEEE, New York, pp 709–712. doi:10.1109/ICIP.2004.1418853
24. Boutell M, Luo J (2005) Beyond pixels: exploiting camera metadata for photo classification. Pattern Recognit 38(6):935–946. doi:10.1016/j.patcog.2004.11.013
25. Lai CL, Chen YS (2009) The application of intelligent system to digital image forensics. In: Proceedings of the international conference on machine learning and cybernetics, vol 5. IEEE, New York, pp 2991–2998. doi:10.1109/ICMLC.2009.5212589
26. Hu Y, Li CT, Zhou C (2010) Selecting forensic features for robust source camera identification. In: Proceeding of the 2010 international computer symposium (ICS), pp 506–511. doi:10.1109/COMPSYM.2010.5685458
27. Ozparlak L, Avcibas I (2011) Differentiating between images using wavelet-based transforms: a comparative study. IEEE Trans Inf Foren Secur 6(4):1418–1431
28. Liu Q, Li X, Chen L, Cho H, Cooper AP, Chen Z, Qiao M, Sung AH (2012) Identification of smartphone-image source and manipulation. In: Jiang H, Ding W, Ali M, Wu X (eds) Advanced research in applied

artificial intelligence. Lecture notes in computer science, vol 7345. Springer, Berlin, pp 262–271. doi:10.1007/978-3-642-31087-4-28
29. Sandoval Orozco AL, Rosales Corripio J, Garcia Villalba LJ, Hernandez-Castro JC (2013) Source identification for mobile devices, based on wavelet transforms combined with sensor imperfections. Computing 96(9):829–841
30. Jang CJ, Lee JY, Won Lee J, Cho HG (2007) Smart management system for digital photographs using temporal and spatial features with EXIF metadata. In: Proceedings of the 2nd international conference on digital information management. ICDIM '07, vol 1, pp 110–115. doi:10.1109/ICDIM.2007.4444209
31. Fan J, Kot A, Cao H, Sattar F (2011) Modeling the EXIF-image correlation for image manipulation detection. In: Proceedings of the 18th IEEE international conference on image processing (ICIP), pp 1945–1948. doi:10.1109/ICIP.2011.6115853
32. Boutell M, Luo J (2004) Photo classification by integrating image content and camera metadata. In: Proceedings of the 17th international conference on pattern recognition, vol 4, pp 901–904. doi:10.1109/ICPR.2004.1333918
33. Tesic J (2005) Metadata practices for consumer photos. IEEE Multimedia 12(3):86–92. doi:10.1109/MMUL.2005.50
34. Romero NL, Chornet VG, Cobos JS, Carot AS, Centellas FC, Mendez MC (2008) Recovery of descriptive information in images from digital libraries by means of EXIF metadata. Library Hi Tech 26(2):302–315. doi:10.1108/07378830810880388
35. Choi KS, Lam EY, Wong KY (2006) Automatic source camera identification using the intrinsic lens radial distortion. Optics Express 14(24):11551–11565. doi:10.1364/OE.14.011551
36. Long Y, Huang Y (2006) Image based source camera identification using demosaicking. In: Proceedings of the IEEE 8th workshop on multimedia signal processing, pp 419–424. doi:10.1109/MMSP.2006.285343
37. Dirik AE, Sencar HT, Memon N (2007) Source camera identification based on sensor dust characteristics. In: Workshop on signal processing applications for public security and forensics. IEEE, New York, pp 1–6
38. Tsai MJ, Lai CL, Liu J (2007) Camera/mobile phone source identification for digital forensics. In: Proceedings of the international conference on acoustics speech and signal processing, pp II-221–II-224. doi:10.1109/ICASSP.2007.366212
39. Van LT, Emmanuel S, Kankanhalli M (2007) Identifying source cell phone using chromatic aberration. In: Proceedings of the IEEE international conference on multimedia and expo, pp 883–886. doi:10.1109/ICME.2007.4284792
40. Celiktutan O, Sankur B, Avcibas I (2008) Blind identification of source cell-phone model. IEEE Trans Inf Foren Secur 3(3):553566. doi:10.1109/TIFS.2008.926993
41. Sandoval Orozco A, Arenas Gonzalez D, Rosales Corripio J, Garcia Villalba L, Hernandez-Castro J (2013) Techniques for source camera identification. In: Proceedings of the 6th international conference on information technology, pp 1–9
42. Hamilton E (2004) JPEG file interchange format. Version 1.02, September 1, 1992. http://www.w3.org/Graphics/JPEG/jfif3.pdf
43. Adobe developers association. TiFF Revision 6.0 (1992). http://partners.adobe.com/public/developer/en/tiff/TIFF6.pdf
44. Committee S (2010) Exchangeable image file for digital still cameras: Exif version 2.3, April 26, 2010. http://www.cipa.jp/english/hyoujunka/kikaku/pdf/DC-008-2010-E.pdf
45. Sandoval Orozco A, Arenas González D, Garca Villalba L, Hernndez-Castro J (2015) Analysis of errors in Exif metadata on mobile devices. Multimed Tools Appl 74(13):4735–4763. doi:10.1007/s11042-013-1837-6
46. International T, Telecommunications P, Press C, Illustrator A (2007) International Press Telecommunications Council. Image Rochester, pp 3–5
47. Wandel M (2010) Exif Jpeg header manipulation tool. http://www.sentex.net/~mwandel/jhead/
48. Adobe Developers Association (2005) XMP specification. http://partners.adobe.com/public/developer/en/xmp/
49. Harvey P (2005) ExifTool. http://www.sno.phy.queensu.ca/~phil/exiftool/
50. Raskin A (2007) Exif viewer 1.81. https://addons.mozilla.org/es-es/firefox/addon/exif-viewer/
51. Kowalski M (2010) ExifPro image viewer. http://www.exifpro.com/

ARTICLE IN PRESS

Future Generation Computer Systems ▮ (▮▮▮▮) ▮▮▮–▮▮▮

Contents lists available at ScienceDirect

Future Generation Computer Systems

journal homepage: www.elsevier.com/locate/fgcs

A PRNU-based counter-forensic method to manipulate smartphone image source identification techniques

Luis Javier García Villalba [a,*], Ana Lucila Sandoval Orozco [a], Jocelin Rosales Corripio [a], Julio Hernandez-Castro [b]

[a] Group of Analysis, Security and Systems (GASS), Department of Software Engineering and Artificial Intelligence (DISIA), Faculty of Computer Science and Engineering, Office 431, Universidad Complutense de Madrid (UCM), Calle Profesor José García Santesmases s/n, Ciudad Universitaria, 28040 Madrid, Spain

[b] School of Computing, Office S129A, University of Kent, Cornwallis South Building, Canterbury CT2 7NF, UK

HIGHLIGHTS

- This research presents a Counter-Forensic method based on sensor noise and wavelet transform.
- The paper proposes two algorithms, one to destroy of image identity and another to forge of a given image identity.
- The aim of destroy of image identity algorithm is anonymize an image.
- The aim of forge of a given image identity algorithm is forgery the source of acquisition of an image.

ARTICLE INFO

Article history:
Received 31 August 2015
Received in revised form
6 November 2016
Accepted 8 November 2016
Available online xxxx

Keywords:
Counter forensics, Forensics analysis
Image anonymity
Image forgery
Photo response non uniformity
Wavelet

ABSTRACT

The increased diffusion of digital images generated by mobile devices through social networks, personal and professional communications, etc. is self-evident. This creates potential problems because some of these images may be used as supporting evidence for different criminal cases. In this paper, algorithms are proposed based on sensor noise and wavelet transforms which can alter the information which is usually employed to find the source of an image, and forge it so that it could point to a different, unrelated device. In the state of art we will show that there are already some algorithms capable of carrying out these manipulations, but they generally need much more and more complex data than our proposal. They also generally need physical access to the camera whose generated images you want to tamper. Our proposal algorithm to destruct the image identifiable data, only needs the picture which will be anonymized. Also, our proposal to forge the image identifiable data only needs a set of photos from the attacker camera, and the picture to be tampered. In particular, it does not need access to the camera that will be falsely linked to the picture. These scenarios are the most common and realistic. The algorithms proposed will help to strengthen existing techniques and develop new forensic approaches for mobile image source identification that will be more robust against attacks.

1. Introduction

It has never been so easy to alter images as it is today, given the existence of powerful and sophisticated software for image processing and manipulation. This ease raises questions about the integrity of images and their validity as forensic evidence.

At present, mobile devices (phones, smartphones, PDA's, tablets, etc.) sales are still rising even with the impact of the global financial crisis. Nowadays, it is usual to see how pictures from mobile devices are taken and used for a wide variety of situations (personal life, news, legal evidence, mobile phone applications, etc.) A further consequence of digital images widespread use is that they are used today as silent witnesses in legal proceedings, frequently being a crucial piece of evidence of the crime [1]. Consequently, many areas may benefit from stronger image forensics, such as the fight against child pornography, piracy, and other crimes such as kidnapping.

For these reasons, digital image forensics have become a topic of interest in recent years. Image forensic analysis arises with the idea

* Corresponding author.
E-mail addresses: javiergv@fdi.ucm.es (L.J. García Villalba), asandoval@fdi.ucm.es (A.L. Sandoval Orozco), jocelinr@ucm.es (J. Rosales Corripio), J.C.Hernandez-Castro@kent.ac.uk (J. Hernandez-Castro).

http://dx.doi.org/10.1016/j.future.2016.11.007
0167-739X/

Please cite this article in press as: L.J. García Villalba, et al., A PRNU-based counter-forensic method to manipulate smartphone image source identification techniques, Future Generation Computer Systems (2017), http://dx.doi.org/10.1016/j.future.2016.11.007

of restoring the reliability of digital images which otherwise could be considered very easily modifiable. Just as most fields of study have a countercurrent, in this case, people like spies, criminals or scammers make efforts to manipulate images for their own benefit. They use the knowledge of digital image forensics with the aim of deleting or even supplanting the fingerprints or traces that are used to determine the image source. Many of the forensic algorithms in the literature were not designed to be robust against such behavior, and as a result they are easy to fool. In the same way, image forensic methods may benefit from studies like us on attack techniques, with the purpose of strengthening next generation algorithms.

2. Image acquisition process in a digital camera

To understand digital image forensics it is essential to know in detail about the image acquisition process in digital cameras. This process is summarized in Fig. 1.

In order to generate a digital image, first the lens system collects light from the scene controlling exposure, focus, and image stabilization. Then the light enters the camera through the lens, and it goes through a combination of filters (at least the infrared and anti-aliasing filters) to ensure maximum image quality. In order to produce a color image the *Color Filter Array* (CFA) is used. After that, light is focused onto the imaging sensor that is an array of light-sensing elements called pixels. After light impacts against pixels they generate an analog signal proportional to the intensity of light received, which is converted into a digital signal to be processed by the *Digital Image Processor* (DIP). Finally, the complete final image is formed by the DIP, which performs some operations such as demosaicing, white point correction, gamma correction, compression, etc., aiming to produce a visually pleasing image.

3. Image source acquisition identification techniques

Forensic analysis of digital images can be mainly divided into two branches [2]: tamper detection and image source acquisition identification. The first of these tries to discern if an image has suffered any kind of alteration after its creation, that is, to detect whether the image has been manipulated. The second research area has the aim of identifying the type or class of the device used for image acquisition.

This paper focuses on attacks against one of the main techniques in the field of source identification: sensor noise analysis. Therefore, we will briefly show aspects about image source acquisition identification and present the state of the art on the use of sensor noise.

For designing techniques and algorithms in any of these areas we will take advantage of some special features found on images created by mobiles, features that serve as a fundamental basis for forensic analysis. In [3,4] a study of the multiple features that may be subject to forensic analysis on mobile devices is presented.

Regarding the area of source identification, the studies conducted so far are basically divided into four groups depending on the type of information they use for identification [3]: those based on the lens system aberrations, the ones based on the CFA choice and the specification of the color interpolation algorithms, those based on image characteristics (color, quality metrics and frequency domain) and the ones that employ on sensor noise. In addition to the above there is another group of techniques based on metadata.

Digital cameras have a powerful source of information which is the embedded metadata in digital images files. These metadata provide relevant information to supplement the main content of a digital document. Techniques based on the image metadata are the simplest. There are plenty of studies focused on the different types of metadata, both for finding information and for image classification [5–8]. Metadata can also be used as input or aid for other forensic techniques. For instance, in the application of content-based image techniques, *Exchangeable image file format* (Exif) metadata can provide a large number and variety of technical information, which may allow an increase in the success rates or improve the results of the application of certain forensic algorithms [9–11]. However, these techniques depend largely in the metadata inserted by manufacturers when the image is created and the correction. In [12] authors make an in-depth study on this topic. Moreover, this method is the most vulnerable to malicious alterations.

The techniques based on sensor noise are mainly divided into two categories: pixel defects and *Sensor Pattern Noise* (SPN). The first one studies the pixel defects, hot pixels, dead pixels, the row or column defects and the group defects. In the second technique a pattern is constructed by averaging multiple residual noises obtained by a noise removal filter, and the presence of the pattern is determined by applying a classification method such as correlation or *Support Vector Machine* (SVM).

Geradts et al. [13] examine *Charge Coupled Device* (CCD) pixel defects but their approach is not fully relevant in our case *Complementary Metal Oxide Semiconductor* (CMOS), which will be later explained. This technique includes point defects, hot points, dead pixel, pixel traps and cluster defects. They noted that each one of the cameras had a different defect pattern. Nevertheless, they also concluded that the number of defects differed between pictures and varies greatly depending on the contents of the image and temperature. Finally, the study found that cameras with a high-end CCD did not have this kind of problem at all, meaning that not all cameras suffered from this issue. It is also true that most cameras have additional mechanisms to compensate for this kind of problem.

Lukas et al. [14] propose a method based on the *Pixel Non-Uniformity* (PNU), which is a great source for the retrieval of noise patterns, allowing them to identify sensors and therefore the source camera. However, the result for pictures of different sizes and for cropped images was not satisfactory [3].

Costa et al. [15] postulate an approach for source camera attribution considering an Open Set scenario, which means that full access to all possible source cameras cannot be taken for granted. This proposal comprises three strands: definition of regions of interest, feature characterization, and source camera attribution. Different regions of the images can contain different information about the fingerprint of the source camera. This approach, in contrast to others, considers different areas of interest and not just the central region of the image. It is assumed that these regions coincide with the principal axis of the lens and should have more scene details because amateur photographers usually focus the object of interest in the center of the lens. An important aspect to note from this kind of region characterization is that it allows the comparison of images with different resolutions without color interpolation artifacts, and it is not necessary to do zero-padding. The technique of this paper uses the SPN of the images and it extracts a set of 36 features which are used for further classification using SVM.[16] is an extension of [15], where new image source acquisition identification experiments are carried out using a much larger number of cameras (400) in comparison with other state of the art works.

In [17] authors study recent developments in the field and proposes the mixture of two techniques (sensor imperfections and wavelet transforms) to get better source identification of images generated with mobile devices.

In [18] a method for image source acquisition identification based on *Photo Response Non Uniformity* (PRNU) noise features extraction, which uses SVM machine for classification, is presented. This work is based specifically on mobile devices images.

Please cite this article in press as: L.J. García Villalba, et al., A PRNU-based counter-forensic method to manipulate smartphone image source identification techniques, Future Generation Computer Systems (2017), http://dx.doi.org/10.1016/j.future.2016.11.007

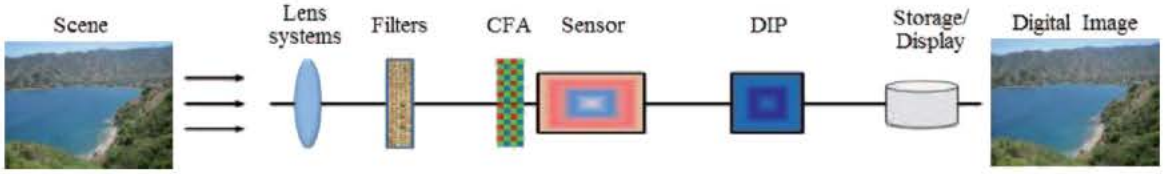


Fig. 1. Image acquisition process in a digital camera.

Authors in [19] use the pattern noise of an imaging sensor to classify digital photographs according to the source smartphone from which they originated.

In [20] authors present a proposal for clustering of images according to their source acquisition based on the combination of hierarchical and flat clustering and the use of SPN noise.

4. Attacks on digital image forensics

In contrast to the prominent role of digital images in our society today, research in the field of image authenticity is still in a very preliminary stage. Most publications in this emerging field still lack rigorous and robust discussions against strategic counterfeiters, who anticipate and try to fool the use of forensic techniques [2].

The area in charge of studying attacks on imaging forensic techniques is known as counter-forensics. Attacks on digital image forensic algorithms are aimed at systematically confusing or misleading the procedures for identifying the source of an image or detecting malicious image manipulations. These attacks could have one of the following goals: camouflage malicious post-processing of images or manipulating the image source identification.

With respect to the aim of manipulating the image source identification, as well as for the process of source identification, the sensor noise extracted from images is generally used. A logic counter for this technique consists in removing all the sensor noise. Taking a step further, one could also think of the possibility of removing the sensor noise and replacing it with the sensor noise belonging to another camera. Therefore this goal can be divided into two branches: destruction of image identity or forgery of image identity.

Below, some of the general concepts of each of these three goals of counter-forensics will be treated, as well as some proposed solutions.

4.1. The post-processing camouflage

These techniques are designed to hide that different processes have been applied to an image. This is achieved by analyzing the traits left by processes on the image during their application, in order to counter them.

In [21] the dependencies introduced during the resizing or rotating process of digital images are examined in detail.

In [22] authors study the statistical coefficients of *Joint Photographic Experts Group* (JPEG) to detect recompression.

In [23] the phase congruence is analyzed to detect image composition done by cutting and pasting parts of different images.

In [2] a proposal to hide resampling, which is the process of resizing images with interpolation, and extremely common in operations like scaling and rotating images. Resampling detector algorithms are based on the search for periodic and systematic dependencies between neighboring pixels, as these are inserted when applying the resampling operation. To hide resampling it is necessary to break the periodic equidistance introducing geometric distortions, also known as watermark attacks. In this case, a random distortion vector is overlapped in each pixel position where a parameter determines the distortion degree introduced. To avoid creating visible noise features in the image, the distortion strength should be modulated using two edge detectors, one vertically and the other horizontally.

In [24] the features introduced in the JPEG compression process are studied and a method for detecting JPEG traces is proposed, even when anti-forensics aspects are taken into account in the compression.

In [25] a method to detect image splicing is proposed. Finally, in [26] a complete and detailed survey about passive image forgery detection techniques can be found.

4.2. Destruction of image identity

In [2] it was shown that the simple subtraction of wavelet domain characteristics of the images is not sufficient to eliminate noise in an image, and also that this procedure leaves visible traces on the resulting image. There is another well-known method for removing noise from an image called flatfielding, and this method is typically used in astronomy or planes scanning, to improve image quality.

The flatfielding is based on the main components of image noise: the *Fixed Pattern Noise* (FPN) and the PRNU. There are several sources of imperfections and noise introduced at different stages of the creation pipeline of an image. Even if an uniform and fully lighted picture is taken, it is still possible to see small changes in the intensity between pixels. This is due to the shot noise being random and, in large part, the pattern noise being deterministic and being approximately equal if several pictures of the same scene are taken.

The noise pattern of an image refers to any spatial pattern that does not change from one image to another. This noise is composed of the spatial noise, which is independent of the signal named FPN; and also of the difference in the response of each pixel to the incident signal, known as PRNU. The noise pattern structure is shown in Fig. 2.

Noise FPN is generated by the dark current and it also depends on exposure and temperature. Since the fixed noise pattern is an independent additive noise, some cameras automatically remove it by subtracting a dark frame to the generated images. Noise PRNU is the dominant part of the sensor noise pattern of an image, and it is a multiplicative noise dependent. Noise PRNU is mainly formed by the non-uniformity of pixel PNU and by the low frequency defects as *zoom* settings and light refraction in the dust particles and lenses. Noise PNU is the light sensitivity difference between pixels of the sensor array. It is generated by the lack of homogeneity of the silicon wafers and by the imperfections during the sensor manufacturing process. Due to its nature and origin, it is very unlikely that even sensors from the same wafer will have PNU correlated patterns. This noise is not affected by ambient temperature nor by humidity. Noise PNU is usually more common, complex and significant in CMOS sensors, due to the complexity of pixel array circuitry.

The FPN noise is calculated in terms of a dark frame d (Eq. (1)) averaging N images x_{dark} taken in a completely dark environment that can be emulated by completely covering the camera lens. That is to say, the Eq. (1) computes the dark frame d.

$$d = \frac{1}{N}\sum x_{dark}. \tag{1}$$

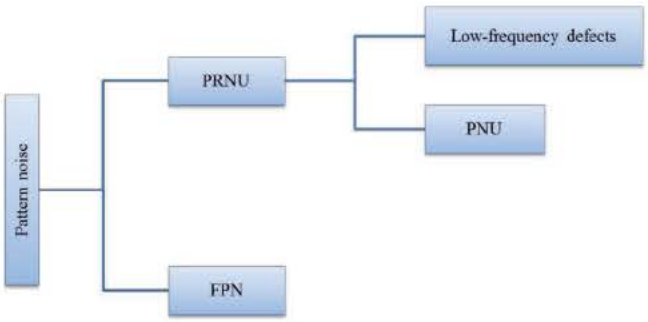


Fig. 2. Sensor noise pattern.

The PRNU noise is calculated in terms of a flatfield f (Eq. (2)) averaging L images $x_{lighted}$ from homogeneously lighted scenes. It is necessary to eliminate the noise FPN from the L images by subtracting the dark frame d before computing the average.

$$f = \frac{1}{L}\sum_{L}\left(x_{lighted} - d\right). \tag{2}$$

As described in [14,2], attackers may try to avoid the correct source identification since it is possible to delete and to remove image fingerprints. The fingerprint subtraction from an image x taken with a specific camera is computed with the Eq. (3) subtracting a dark current d to the original image x and then dividing the result by the flat frame f. $\tilde{x}$ is the image with its fingerprint subtracted.

$$\tilde{x} = \frac{x - d}{f}. \tag{3}$$

Despite the fact that the results obtained with this technique are good, it has some drawbacks:

- Performing a perfect flatfielding with a large number of photos is difficult because the parameters to compute the PRNU and FPN must match the parameters from the victim picture.
- The proposal assumes that the attacker can access the source camera of the image x to generate the dark and the flat frames, this scenario is not close to reality.

There are other less robust possibilities to destroy image identity which in some cases may be effective because they do not need extra images from the source camera. However, instead of this facility the image quality could be reduced and some visual features could be introduced into images. Examples of this kind of technique are: rotating the image a few degrees, scaling the image or applying a Gaussian filter that blurs the image.

In [27] PRNU noise is used to pinpoint the camera device which could be undesirable for some users who want to protect their privacy and preserve their anonymity while sharing or spreading images.

In [28] the authors provide an analysis of the seam-carving-based source camera anonymization method by determining the limits of its performance introducing two adversarial models. The results of the analysis shows that the effectiveness of the deanonymization attacks depend on various factors that include the parameters of the seam-carving method, strength of the PRNU noise pattern of the camera, and an adversary's ability to identify uncarved image blocks in a seam-carved image.

In [29] a technique for circumventing the PRNU based source attribution by mainly focusing on adaptive PRNU denoising method and seam-carving based anonymization is evaluated. Moreover, a panoramic-image-stitching as a means to impede source attribution is introduced.

In [30] an improvement on the existing adaptive PRNU denoising method against source camera identification is introduced and anonymization benchmarks with other source anonymization techniques are provided.

4.3. Forgery of image identity

In the same way that image noise can be removed using the flatfielding technique, it is possible to inject the sensor noise from a different camera using the inverse flatfielding with Eq. (4) [2].

$$\tilde{y} = \tilde{x} \cdot f_{forged} + d_{forged} \tag{4}$$

where f_{forged} and d_{forged} correspond to the camera that is intended to attack and $\tilde{x}$ is the original image without noise.

In [31] the Algorithm 1 is proposed to forge the identity of a camera, where **C1** is the attacker camera, **C2** is the victim camera and **P** is a picture taken by **C2**.

Algorithm 1: Forgery of Image Identity

1. Compute attacker camera **C1** fingerprint average $F(\mathbf{C1})$;
2. Take a picture **P** with the victim camera **C2**;
3. Add F(**C1**) to the picture **P**;

In case the dimensions of $F(\mathbf{C1})$ and $\boldsymbol{P}$ do not match, a cut or a reconstruction must be applied to match the image sizes. An improvement to the previous falsification algorithm is also proposed by [31] to mask the features of the camera **C2**. This technique is presented in Algorithm 2.

Algorithm 2: Forgery of Image Identity with Concealed Camera

1. Compute attacker camera **C1** fingerprint average $F(\mathbf{C1})$;
2. Compute victim camera **C2** fingerprint average $F(\mathbf{C2})$;
3. Take a picture **P** with the victim camera **C2**;
4. Subtract $F(\mathbf{C2})$ to **P**;
5. Add F(**C1**) to the picture **P**;

The subtraction of $F(\boldsymbol{C2})$ tries to eliminate the correlation between picture $\boldsymbol{P}$ and the camera $\boldsymbol{C2}$, that is to say the existing fingerprint is subtracted before applying the attacker camera fingerprint.

[32] proposes a technique based on the study of second-order statistics derived from the co-occurrence matrix for detect the presence of counter-forensic attacks.

[33] proposes an image forgery detection scheme that identifies a tampered foreground or background image using image watermarking and alpha mattes, the proposed method uses (a) component hue difference based spectral matting, (b) image watermarking based on the discrete wavelet transform, discrete cosine transform, and singular value decomposition and (c) the difference between the obtained singular values are used to detect tampering of foreground or background image.

5. Proposed PRNU-based counter-forensic method

In this section we propose two algorithms, one to destroy the image identity and another to forge a given image identity. The aim of the first algorithm is anonymize an image, or in other words, remove as much as possible any trace that allows the source image acquisition identification by a forensic analyst. The aim of the second algorithm is forgery the source of acquisition of an image, or in other words, doing unsuccessful a forensic analysis which has the aim of identify the image source acquisition and make its result the falsified source acquisition.

In [18] it is determined that the use of the sensor pattern noise (PRNU) together with the wavelet transform is an effective method for source identification, reaching an average success rate of 87.21%. An estimated counter-forensic technique against this type of identification may be based on these elements. Therefore, the presented algorithms base their working on PRNU noise handling and wavelet transform.

Please cite this article in press as: L.J. García Villalba, et al., A PRNU-based counter-forensic method to manipulate smartphone image source identification techniques, Future Generation Computer Systems (2017), http://dx.doi.org/10.1016/j.future.2016.11.007

5.1. Proposed algorithm of destruction of image identity

In this section we propose an algorithm based on [34] to extract and to remove the sensor fingerprint from an image P_1. The algorithm proposed obtains a features vector for classification purposes. The proposed algorithm, on the other hand, has the aim of obtaining an image with its identity destroyed removing the PRNU noise.

Among different filters that exist for eliminating noise from images those using the wavelet transform work better because the residual noise obtained with this kind of filter contains the fewest features from the scene. Generally, the areas around the edges are misinterpreted when a less robust noise removal filter is only used, such as the Wiener filter or median filter. For this reason we selected the noise removal filter based on wavelet transform in combination with Wiener filter. For each wavelet decomposition level we obtain the high-frequency components H (horizontal), V (vertical) and D (diagonal). The Algorithm 3 describes the steps to remove the sensor fingerprint.

Where I_{clean} is obtained by applying some elimination filter as described in Section 4.2. Particularly in this work the noise elimination is performed by applying the Algorithm 3. Note that the I_{clean} obtained is not exactly a picture without any noise since the sensor noise pattern is formed by the PRNU and PNU as shown in Section 4.2. The Algorithm 3 only removes the PRNU noise of the I_{clean} and FPN does not. However for ease in the naming and nomenclature will be used I_{clean}, to name the image without PRNU noise.

5.2. Proposed algorithm of forgery of image identity

In this section we propose an algorithm to forge the sensor pattern noise from a camera C_1 to an image P_2 belonging to a camera C_2 without requiring access to camera C_2. This algorithm uses Algorithm 3 previously presented in Section 5.1.

Algorithm 3: Removing the PRNU

Input: I the victim image

```
1  procedure REMOVEPRNU(I)
2    Apply a wavelet decomposition in 4 levels to I;
3    foreach wavelet decomposition level do
4      foreach wavelet component c ∈ {H, V, D} do
5        Compute the local variance;
6        if adaptive variance then
7          Compute 4 variances with windows
             of size: 3, 5, 7 and 9 respectively;
8          Select the minimum variance;
         else
9          Compute the variance with a window
             of size 3;
10     Compute noiseless wavelet components
         applying the Wiener filter to the variance;
11   Obtain I_clean by applying the inverse wavelet transform
     with clean components calculated;
12 end procedure
```

Source identification Techniques based on PRNU estimate the sensor fingerprint of images using the equation:

$$I_{noise} = I - I_{clean}. \tag{5}$$

The pattern noise PNU is computed by averaging the residual noise of several images with the following equation:

$$P_{noise} = \frac{1}{N}\sum_{i=1}^{N} I_{noise}. \tag{6}$$

Once it is possible to remove sensor noise and to extract the sensor pattern noise, the image identity falsification could be envisaged. The Algorithm 4 shows the steps to follow to fake the image identity.

Algorithm 4: Forging the PRNU

Input: I the victim image
N the number of flat images from forger camera

```
1 procedure FORGEIMG(I, N)
2   I_clean ← REMOVEPRNU(I);
3   P_noise ← EXTRACTPRNU(N);
4   Apply a wavelet decomposition in 1 levels to
      I_clean obtaining components L_I, H_I, V_I and D_I;
5   Apply a wavelet decomposition in 1 levels to
      P_noise obtaining components H_P, V_P and D_P;
6   Compute the forged wavelet components
      with c_F=c_I+c_P where c∈{H,V,D};
7   Obtain I_forged applying the inverse wavelet
      transform with L_I,H_F,V_F y D_F;
8 end procedure
```

In order to have a better quality pattern and to obtain better results in forgery a number of images N bigger than 50, are recommended according to experiments, also the images have been acquired from non-textured uniformly lighted flat surfaces, as flat surfaces could be considered pictures of clear sky or white paper.

6. Experimental setting

This section describes the experiments performed with the algorithms for removing the sensor fingerprint (Algorithm 3) and for forging the source camera fingerprint (Algorithm 4).

The experiments on elimination and falsification of sensor fingerprints were performed using the proposed implementations and the tool "PRNU Decompare" [35], which uses the flatfielding technique described in Section 4.2 and allows the elimination and falsification of sensor pattern noise. This tool requires a picture of a dark frame as input and a number N of images of flat surfaces uniformly lighted (a minimum of 30 frames is recommended). We compared the results of our proposal algorithms with the obtained results in the experiments. For this purpose we used the tool "NFI PRNU Compare" [36], this tool allows us to compare images and sensor pattern noises from various images. "NFI PRNU Compare" uses as a measure to compare the correlation, which is employed in many other works such as [37,15,38] among others, to compare images and noise patterns. It is important to note that our proposal does not need a set of images from the camera with the identity we want to destroy from the picture, we only need the image itself. Also, we do not need any set of photos or have access to the victim camera in the case of forgery of the identity, we only need a set of pictures of the attacker camera. That is, our proposed algorithm requires less input images than "PRNU Decompare" to do the same function.

Please cite this article in press as: L.J. García Villalba, et al., A PRNU-based counter-forensic method to manipulate smartphone image source identification techniques, Future Generation Computer Systems (2017), http://dx.doi.org/10.1016/j.future.2016.11.007

Table 1
Comparison between patterns and noiseless images.

Pattern	Picture	Red	Green	Blue	Sum
LG E510f	Original	−0.014645672	−0.0017777978	−0.007864626	−0.024288096
	Proposal	−0.015506644	−0.003044259	−0.008411303	−0.026962206
	Decompare	−0.018929206	−0.0023383496	−0.012027217	−0.033294775
LG E400	Original	0.011481647	0.010190065	0.01825918	0.039930895
	Proposal	0.010315191	0.008225861	0.017940063	0.036481116
	Decompare	0.010638827	0.009045472	0.016430777	0.036115076
Nokia 800 Lumia	Original	0.011352311	0.011754888	0.019119238	0.042226437
	Proposal	0.009912337	0.009113852	0.016991276	0.036017465
	Decompare	0.010812875	0.009995133	0.014978331	0.035786339
Apple iPhone 5	Original	−0.015712192	−0.002311284	−0.007307031	−0.025330507
	Proposal	−0.016984729	−0.003462913	−0.009599754	−0.030047396
	Decompare	−0.019395503	−0.003140395	−0.009915681	−0.032451579
Sony ST25i	Original	−0.012013772	−0.002127295	−0.006817536	−0.020958603
	Proposal	−0.014839721	−0.003142763	−0.009114359	−0.027096843
	Decompare	−0.016545112	−0.002611324	−0.011200112	−0.030356548
Samsung GTI-9000	Original	0.017310016	0.010754888	0.016119238	0.044184142
	Proposal	0.014015311	0.007826601	0.014491394	0.036333306
	Decompare	0.014979864	0.007992510	0.012984029	0.035956403

Table 2
MAE and MSE of the images and destructed identity images.

Image	MAE			MSE		
	Red	Green	Blue	Red	Green	Blue
Nokia 800 Lumia original	0.8410	0.8183	0.8428	2.6498	2.1777	2.3869
Nokia 800 Lumia destructed identity	0.8368	0.8135	0.8388	2.6245	2.1531	2.3617
Sony ST25i original	0.8976	0.8705	0.8916	4.0664	3.3754	3.7108
Sony ST25i destructed identity	0.8924	0.8656	0.8869	4.0324	3.3465	3.6783

6.1. Destruction of image identity

For this experiment photos from 6 digital cameras (LG E510f, LG 400, Nokia 800 Lumia, Sony ST25i (Xperia U), Apple iPhone5, Samsung GTI-9000) were used. From each device 50 photos of uniformly lighted flat images were taken, and one totally dark picture was generated by completely covering the camera lens. One picture was randomly selected from the photo database of each camera to be used for the removal of the image identifiable data. All photos were cropped to a size of 1024 x 1024 pixels.

Initially, a first set of images without fingerprint was generated using the Algorithm 3. It should be noted that the noise elimination only required the picture which sought to eliminate the sensor noise and not additional ones. Then, a second group of images without fingerprint was generated using the tool "PRNU Decompare", using 50 flat images and a dark frame image as input to this program. Therefore we have two images without sensor camera fingerprint for each camera, one generated with the Algorithm 3 and one with "PRNU Decompare".

For evaluating the effectiveness of the algorithm for elimination of sensor fingerprint six sets of images were compared using the tool "NFI PRNU Compare". With 40 images chosen randomly from the 50 of each camera, "NFI PRNU Compare" gets the noise pattern of each camera, as it is indicated in the recommendations of the tool documentation.

Table 1 shows the results of comparing each sensor pattern noise generated by "NFI PRNU Compare" of each camera with the noiseless images generated by the two tools, and also against the original photograph. "NFI PRNU Compare" allows us to measure how far the patterns compared are similar to each other, the rows closer to the pattern that is being compared are the most similar to it. Also, Table 1 shows the correlation coefficients for each color channel. The closer the value to 1 is considered a high correlation with a high degree of similarity between two linear patterns, a value of 0.5 represents a weak correlation and a negative value indicates a negative correlation in which the increase of a value involves decrease the other. In all tests the noiseless images generated with "PRNU Decompare" resulted to be least similar to the pattern, this was expected as they considered much more information to remove the fingerprint. In all the cases, the comparison of the noiseless images had very similar results, indicating that in this case the proposed algorithm had a good performance getting close to the result of "PRNU Decompare" without the need to use a dark frame nor the 50 flat images.

An important issue to address is whether the image whose identity has been destroyed reduces its image quality or if effects visible to the human eye exist on the image scene. For it, in Fig. 3 two examples of images of Nokia 800 Lumia and Sony ST25i (Xperia U) devices are shown with their respective destroyed identity images. As it can be seen image scenes whose identity has been destroyed do not reveal visual changes.

Even so, it was decided to use *Image Quality Metrics* (IQM) to objectively assess the loss of image quality whose identity has been destroyed with respect to the original.

For this, it was decided to use the Minkowski metrics [39]. In particular we use the Minkowski metric for $\gamma = 1$ which corresponds to *Mean Absolute Error* (MAE) and $\gamma = 2$ with the *Mean Square Error* (MSE). In both cases, high values of MAE or MSE correspond with low quality images. These metrics are applied to each of the RGB bands separately, so that three metrics for the MAE and another three for the MSE are obtained. The values of the quality metrics for each of the bands of the 4 pictures are shown in Table 2, as it can be seen the destruction of image identifiable attributes changes very little in the image quality indexes thus presented. Although a very small reduction is perceived nearly everywhere, it is important to note that in reality the original images will likely be not available, and the changes are so minimal that will not allow to distinguish original or forged images solely based on them.

Please cite this article in press as: L.J. García Villalba, et al., A PRNU-based counter-forensic method to manipulate smartphone image source identification techniques, Future Generation Computer Systems (2017), http://dx.doi.org/10.1016/j.future.2016.11.007

(a) Nokia 800 Lumia: Original image.

(b) Nokia 800 Lumia: Destructed identity image.

(c) Sony ST25i: Original image.

(d) Sony ST25i: Destructed identity image.

Fig. 3. Images and destructed identity images.

Table 3
Devices used for identity forgery.

Forger camera	Victim 1	Victim 2
LG E510f	LG-400	Samsung GT-I8160P
Forger camera	Victim 3	Victim 4
Nokia 800 Lumia	Sony ST25i	Samsung GTI-9000

6.2. Forgery of image identity

For this experiment the same picture set of the experiment Section 6.1 was used. In a similar way to the previous experiment, a camera sensor fingerprint was extracted, then this fingerprint was injected into pictures from another two cameras using the proposed algorithm and "PRNU Decompare", and finally the results were compared with "NFI PRNU Compare". The roles played by each camera are shown in Table 3.

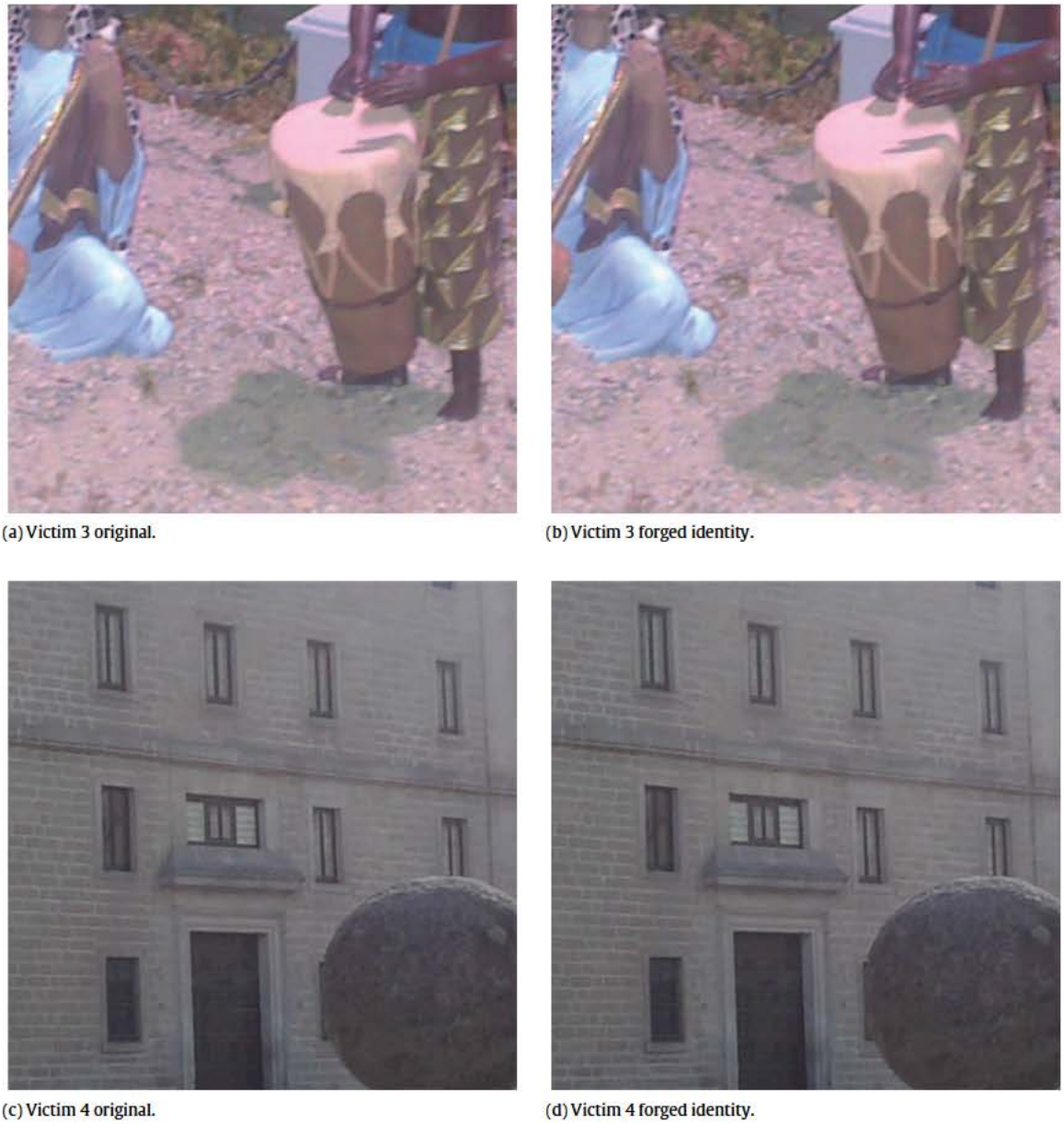
To forge the sensor pattern noise with the proposed algorithm only 50 images from the forger camera were required. For the falsification with "PRNU Decompare" 50 pictures were required as well as one dark frame belonging to the forger and to the victim camera, respectively. After performing forgery in 4 victim cameras the results were compared as summarized in Table 4.

Also, this table shows correlation coefficients for each color channel with respect to the pattern of the forger camera as in the last experiment.

For victims 1, 3 and 4 it can be observed that the three forgeries have a greater similarity with the pattern of the forger camera and that the result of "PRNU Decompare" is closer than the proposal, even though this difference is not significant considering that they use a far greater number of images as a source of information. In the case of victim 2 the result with the proposed algorithm has the least similarity to the forged camera pattern.

The results obtained so far were as expected, because the algorithm proposed in this paper does not assume having access to the victim camera as the tool "PRNU Decompare" does. However, in all the cases the results of our proposal are close to the results of "PRNU Decompare". It is important to notice that in real scenarios, it is not normally possible to access the victim camera. We just need a set of photos of the attacker camera, as "PRNU Decompare"

Please cite this article in press as: L.J. García Villalba, et al., A PRNU-based counter-forensic method to manipulate smartphone image source identification techniques, Future Generation Computer Systems (2017), http://dx.doi.org/10.1016/j.future.2016.11.007



(a) Victim 3 original.

(b) Victim 3 forged identity.

(c) Victim 4 original.

(d) Victim 4 forged identity.

Fig. 4. Images and forged identity images.

needs. However we do not need any special type of content scenery of the set of images and "PRNU Decompare" needs it, as we discussed above.

As in the case of destruction of image identity, the same image quality loss study for the forgery example of victims 3 and 4 is performed. In Table 5 image quality indexes of MAE and MSE of the original images and their respective forged ones are shown. The results can be seen visually in Fig. 4, observing that changes are imperceptible to the human eye.

As can be seen in Table 5 the forgery of an image identifiable data does not change significantly any of the image quality indexes presented, and this difference will be almost impossible to notice if the original images where not provided.

7. Conclusions

In this work we propose two new algorithms, one capable of removing all identifiable data from an image, and other that allows its forgery. The two algorithms are based on the use of different types of sensor noise and wavelet transform. Both algorithms, in addition, have the great advantage over existing ones that they need a much smaller and easier to obtain data input, which make them more applicable and closer real-life scenarios.

In particular, the removal algorithm requires only the image from which to wipe the fingerprint, not a set of planar images and a dark frame from the camera as in the tool "PRNU Decompare". We argue the requirements of "PRNU Decompare" are not realistic in most cases, because a large set of photos with specific characteristics are required for its proper functioning. In the case of our proposed image forgery algorithm, it does not need to access the victim camera. In the case of the proposal [30] it is also required as input 50 not-flat images from the same camera which generated the image victim to perform anonymization.

Both algorithms can be viewed as research contributions to the future strengthening of forensic techniques for detecting intentional manipulation. For example, our first algorithm can be very useful in web applications that upload and display images on the Internet (social networks, directories, etc.), as they allow for completely anonymous image uploading.

Please cite this article in press as: L.J. García Villalba, et al., A PRNU-based counter-forensic method to manipulate smartphone image source identification techniques, Future Generation Computer Systems (2017), http://dx.doi.org/10.1016/j.future.2016.11.007

Table 4
Comparison of patterns, original images and victims.

Picture		Red	Green	Blue	Sum
Victim 1	Decompare	0.009219962	0.0054620425	0.009098741	0.023780745
	Proposal	0.007900867	0.0046529872	0.0083521	0.020905953
	Original	0.0073570074	0.004183661	0.0075896666	0.019130334
Victim 2	Decompare	0.01418404	0.013986045	0.013574668	0.041744754
	Original	0.011300047	0.013949845	0.0125216115	0.0377715
	Proposal	0.008964902	0.0066337977	0.004440412	0.020039111
Victim 3	Decompare	0.00811001	0.004442147	0.009333811	0.021885968
	Proposal	0.007201833	0.00419231	0.009112659	0.020506802
	Original	0.006232409	0.003880702	0.007871138	0.017984249
Victim 4	Decompare	0.009913582	0.007213382	0.008788015	0.025914979
	Proposal	0.008271773	0.007621801	0.008141919	0.024035493
	Original	0.008137003	0.005338104	0.00790911	0.021384217

Table 5
MAE and MSE of the images and forged identity images.

Image	MAE			MSE		
	Red	Green	Blue	Red	Green	Blue
Victim 3 original	0.8976	0.8705	0.8916	4.0664	3.3754	3.7108
Victim 3 forged	0.8926	0.8656	0.8872	4.0323	3.3445	3.6787
Victim 4 original	0.7836	0.7800	0.7820	2.3412	2.2525	2.2724
Victim 4 forged	0.7685	0.7639	0.7661	2.2940	2.2062	2.2207

The effectiveness of the proposed algorithms is overall quite good, even though in some cases they do not get results comparable with those of other algorithms, but they generally achieve close and quite acceptable results, with the benefit of drastically reducing the required input data and being more practical and realistic. These algorithms can be useful as a starting point for future improvements that allow similar results to be obtained for other algorithms or tools, emphasizing and taking into account the limited input data needed and that they can work even when not having access to the victim camera in the case of identity image forgery. Also, the application of both algorithms does not cause noticeable visual changes or degradation in the pictures, and does not significantly reduce the image quality.

Acknowledgments

Part of the heavy computations required for this work were performed in EOLO, part of the HPC for Climate Change of the International Campus of Excellence of Moncloa, funded by both MECD and MICINN. This is a contribution to CEI Moncloa.

References

[1] M. Al-Zarouni, Mobile handset forensic evidence: a challenge for law enforcement, in: Proceedings of the 4th Australian Digital Forensics Conference. School of Computer and Information Science, Edith Cowan University, 2006, pp. 1–12.
[2] T. Gloe, M. Kirchner, A. Winkler, R. Bohme, Can we trust digital image forensics? in: Proceedings of the 15th International Conference on Multimedia, ACM Press, 2007, pp. 78–86.
[3] T. Van Lanh, K.S. Chong, S. Emmanuel, M.S. Kankanhalli, A survey on digital camera image forensic methods, in: Proceedings of the IEEE International Conference on Multimedia and Expo, IEEE, 2007, pp. 16–19.
[4] V.L.L. Thing, K.Y. Ng, E.C. Chang, Live memory forensics of mobile phones, Digit. Investig. 7 (2010) S74–S82.
[5] J. Platt, AutoAlbum: Clustering digital photographs using probabilistic model merging, in: Proceedings of the IEEE Workshop on Content-based Access of Image and Video Libraries, IEEE, 2000, pp. 96–100.
[6] M. Boutell, J. Luo, Beyond pixels: Exploiting camera metadata for photo classification, Pattern Recognit. 38 (6) (2005) 935–946.
[7] J. Tesic, Metadata practices for consumer photos, IEEE Multimedia 12 (3) (2005) 86–92.
[8] N.L. Romero, V.G. Chornet, J.S. Cobos, A.S. Carot, F.C. Centellas, M.C. Mendez, Recovery of descriptive information in images from digital libraries by means of EXIF metadata, Lib. Hi Tech 26 (2) (2008) 302–315.
[9] M. Boutell, J. Luo, Photo classification by integrating image content and camera metadata, in: Proceedings of the 17th International Conference on Pattern Recognition, vol. 4, IEEE Computer Society, 2004, pp. 901–904.
[10] J. Fan, A.C. Kot, H. Cao, F. Sattar, Modeling the EXIF-image correlation for image manipulation detection, in: Proceedings of the 18th IEEE International Conference on Image Processing, ICIP 2011, 2011, pp. 1945–1948.
[11] C.J. Jang, J.Y. Lee, J.W. Lee, H.G. Cho, Smart management system for digital photographs using temporal and spatial features with EXIF metadata, in: Proceedings of the 2nd International Conference on Digital Information Management, Vol. 1, 2007, pp. 110–115.
[12] A.L. Sandoval Orozco, D.M. Arenas González, L.J. García Villalba, J.C. Hernandez-Castro, Analysis of errors in exif metadata on mobile devices, Multimedia Tools Appl. 74 (13) (2015) 4735–4763.
[13] Z.J. Geradts, J. Bijhold, M. Kieft, K. Kurosawa, K. Kuroki, N. Saitoh, Methods for identification of images acquired with digital cameras, in: Proceedings on Enabling Technologies for Law Enforcement and Security, in: SPIE-International Society for Optical Engine, vol. 4232, 2001, pp. 505–512.
[14] J. Lukas, J. Fridrich, M. Goljan, Digital camera identification from sensor pattern noise, IEEE Trans. Inf. Forensics Secur. 1 (2) (2006) 205–214.
[15] F.D.O. Costa, M. Eckmann, W.J. Scheirer, A. Rocha, Open set source camera attribution, in: Proceedings of the 25th Conference on Graphics, Patterns and Images, IEEE, 2012, pp. 71–78.
[16] F.D.O. Costa, M. Eckmann, W.J. Scheirer, A. Rocha, Open set source camera attribution and device linking, Pattern Recognit. Lett. 39 (2014) 92–101.
[17] A.L. Sandoval Orozco, D.M. Arenas González, J. Rosales Corripio, L.J. García Villalba, J.C. Hernandez-Castro, Source identification for mobile devices, based on wavelet transforms combined with sensor imperfections, Computing (2013) 1–13.
[18] J. Rosales Corripio, D.M. Arenas González, A.L. Sandoval Orozco, L.J. García Villalba, J.C. Hernandez-Castro, S.J. Gibson, Source smartphone identification using sensor pattern noise and wavelet transform, in: Proceedings of the 5th International Conference on Imaging for Crime Detection and Prevention, ICDP 2013. 2013, pp. 1–6.
[19] A.L. Sandoval Orozco, L.J. García Villalba, D.M. Arenas González, J. Rosales Corripio, J.C. Hernandez-Castro, S.J. Gibson, Smartphone image acquisition forensics using sensor fingerprint, IET Comput. Vis. (2015).
[20] L.J. García Villalba, A.L. Sandoval Orozco, J. Rosales Corripio, Smartphone image clustering, Expert Syst. Appl. 42 (4) (2015) 1927–1940.
[21] A. Popescu, H. Farid, Exposing digital forgeries by detecting traces of resampling, IEEE Trans. Signal Process. 53 (2) (2005) 758–767.
[22] J. Lukas, J. Fridrich, Estimation of primary quantization matrix in double compressed JPEG images, in: Digital Forensic Research Workshop, 2003, pp. 5–8.
[23] W. Chen, Y. Shi, W. Su, Image splicing detection using 2-D phase congruency and statistical moments of characteristic function, in: Proceedings on Security, Steganography, and Watermarking of Multimedia Contents IX, in: SPIE-International Society for Optical Engine, vol. 6505, 2007, pp. 1–8.
[24] G. Valenzise, M. Tagliasacchi, S. Tubaro, Revealing the traces of jpeg compression anti-forensics, IEEE Trans. Inf. Forensics Secur. 8 (2) (2013) 335–349.
[25] G. Liu, J. Wang, S. Lian, Y. Dai, Detect image splicing with artificial blurred boundary, Math. Comput. Modelling 57 (11) (2013) 2647–2659.
[26] G.K. Birajdar, V.H. Mankar, Digital image forgery detection using passive techniques: A survey, Digit. Investig. 10 (3) (2013) 226–245.
[27] R. Bohme, M. Kirchner, Counter-forensics: Attacking image forensics, in: H.T. Sencar, N. Memon (Eds.), Digital Image Forensics, Springer, New York, 2013, pp. 327–366.

Please cite this article in press as: L.J. García Villalba, et al., A PRNU-based counter-forensic method to manipulate smartphone image source identification techniques, Future Generation Computer Systems (2017), http://dx.doi.org/10.1016/j.future.2016.11.007

[28] A. Dirik, H. Sencar, N. Memon, Analysis of seam-carving-based anonymization of images against PRNU noise pattern-based source attribution, IEEE Trans. Inf. Forensics Secur. 9 (12) (2014) 2277–2290.
[29] A. Karaküçük, A.E. Dirik, H.T. Sencar, N.D. Memon, Recent advances in counter PRNU based source attribution and beyond, Proc. SPIE 9409, Media Watermarking, Security, and Forensics 2015, 94090N (March 4, 2015). http://dx.doi.org/10.1117/12.2182458.
[30] A. Karakucuk, A.E. Dirik, Adaptive photo-response non-uniformity noise removal against image source attribution, Digit. Investig. 12 (2015) 66–76.
[31] M. Steinebach, H. Liu, P. Fan, S. Katzenbeisser, Cell phone camera ballistics: Attacks and countermeasures, in: Proceedings on Multimedia on Mobile Devices, in: SPIE-International Society for Optical Engine, 2010, pp. 1–9.
[32] S. Jothimani, P. Betty, Image authentication using global and local features, in: Proceedings of the International Conference on Green Computing Communication and Electrical Engineering, 2014, pp. 1–5.
[33] W.C. Hu, W.H. Chen, D.Y. Huang, C.Y. Yang, Effective image forgery detection of tampered foreground or background image based on image watermarking and alpha mattes, Multimedia Tools Appl. (2015) 1–22.
[34] M. Goljan, J. Fridrich, T. Holotyak, New blind steganalysis and its implications, Proc. SPIE 6072 (2006) 1–13.
[35] Netherlands Forensic Institute. PRNU Decompare. 2013.
[36] Netherlands Forensic Institute. NFI PRNU Compare. 2013.
[37] M. Goljan, J. Fridrich, M. Chen, Defending against fingerprint-copy attack in sensor-based camera identification, IEEE Trans. Inf. Forensics Secur. 6 (1) (2011) 227–236.
[38] A. Uhl, Y. Holler, Iris-sensor authentication using camera prnu fingerprints, in: Proceedings of the 5th IAPR International Conference on Biometrics, (ICB), IEEE, 2012, pp. 230–237.
[39] Y. Hu, C.T. Li, C. Zhou, Selecting forensic features for robust source camera identification, in: Proceedings of the International Computer Symposium. Tainan, China, 2010, pp. 506–511.

Luis Javier García Villalba received a Telecommunication Engineering degree from Universidad de Málaga (Spain) in 1993 and holds a M.Sc. in Computer Networks (1996) and a Ph.D. in Computer Science (1999), both from the Universidad Politécnica de Madrid (Spain). He was a visiting scholar at COSIC (Computer Security and Industrial Cryptography, Department of Electrical Engineering, Faculty of Engineering, Katholieke Universiteit Leuven, Belgium) in 2000 and a Visiting Scientist at IBM Research Division (IBM Almaden Research Center, San Jose, CA, USA) in 2001 and 2002. He is currently Associate Professor at the Department of Software Engineering and Artificial Intelligence of Universidad Complutense de Madrid (UCM) and Head of GASS (Group of Analysis, Security and Systems) Research Group, located in the School of Computer Science at the UCM Campus. His professional experience includes research projects with Hitachi, IBM, Nokia and Safelayer Secure Communications. He has received research funding from two EU H2020 projects. His main research interests are cryptography, coding, information security and its applications.

Ana Lucila Sandoval Orozco received a Computer Science Engineering degree from the Universidad Autónoma del Caribe (Colombia) in 2001. She holds a Specialization Course in Computer Networks (2006) from the Universidad del Norte (Colombia) and a M.Sc. in Research in Computer Science from the Universidad Complutense de Madrid (Spain) since 2009. She is currently a postdoctoral researcher at Universidad Complutense de Madrid (Spain). Her main research interests are coding theory, information security and its applications.

Jocelin Rosales Corripio received a Computer Science Engineering degree from the Benemerita Universidad Autónoma de Puebla (Mexico) in 2008. She holds a M.Sc. in Research in Computer Science from the Universidad Complutense de Madrid (Spain) in 2013. She is currently a Ph.D. Student at the Universidad Complutense de Madrid (Spain) and a Research Assistant at Complutense Research Group GASS. Her main research interests are coding theory, information security and its applications.

Julio Hernandez-Castro received a B.Sc. degree in Mathematics from the Universidad Complutense de Madrid, Spain, in 1995, a M.Sc. degree in Coding Theory and Network Security from the Universidad de Valladolid in 1999, and a Ph.D. degree in Computer Science from Carlos III University of Madrid in 2003. He is currently a Senior Lecturer in Computer Security at the School of Computing, University of Kent, UK. He is also associated with Kent Cyber Security Center. His interests include Cryptology, Steganography & Steganalysis, Computer & Network Security, Computer Forensics, CAPTCHAs, RFID Security, and the application of Non-Standard techniques to Cryptology.

Please cite this article in press as: L.J. García Villalba, et al., A PRNU-based counter-forensic method to manipulate smartphone image source identification techniques, Future Generation Computer Systems (2017), http://dx.doi.org/10.1016/j.future.2016.11.007